

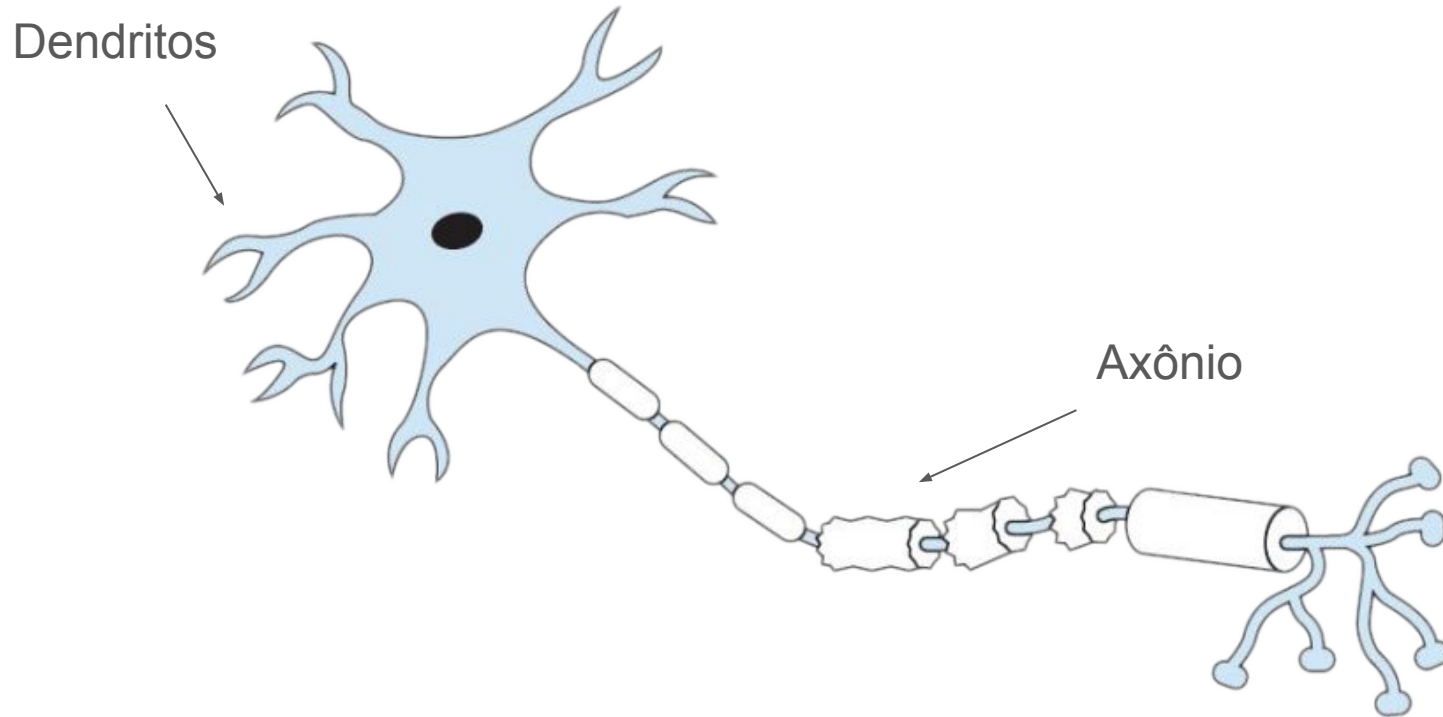
data

Redes Neurais: MLP e Backpropagation

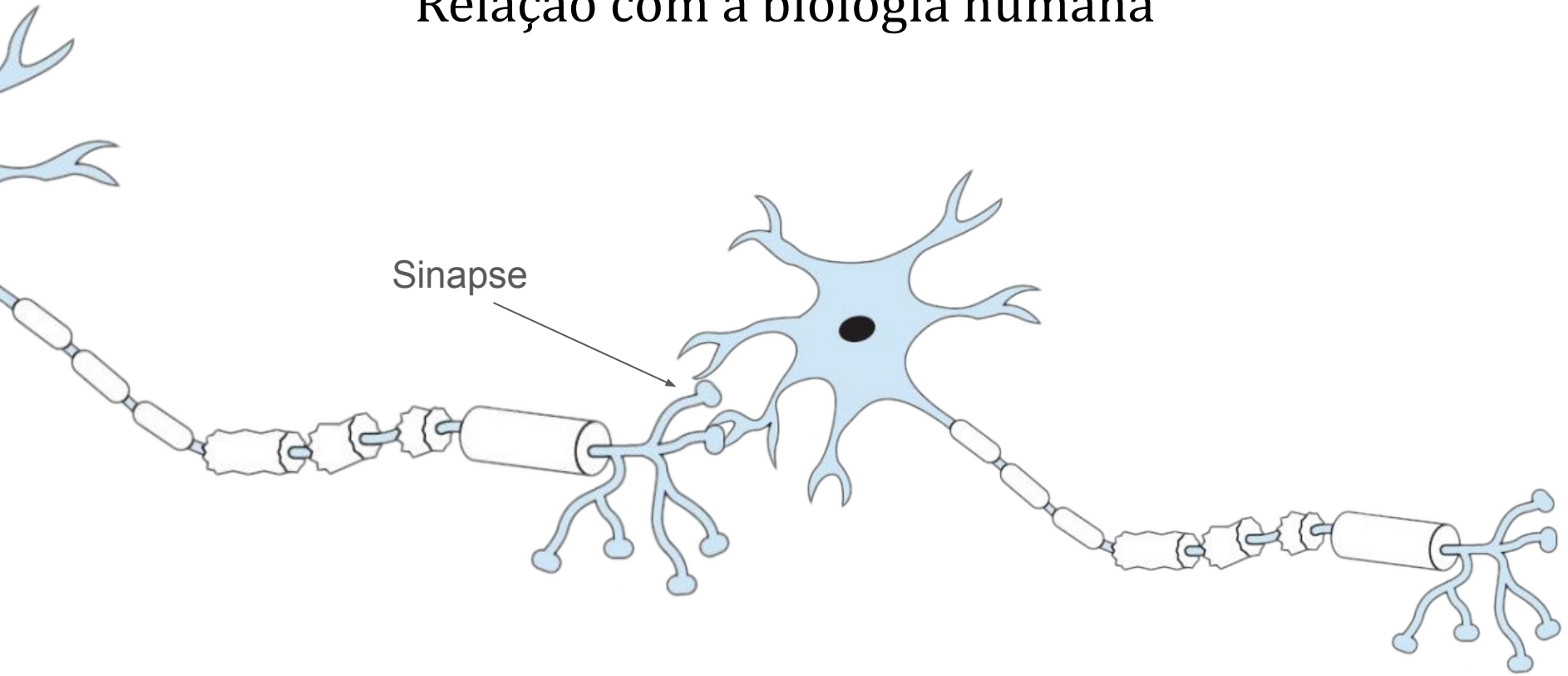
por Lázaro Vinaud e
Vinicio Yusuke



Relação com a biologia humana



Relação com a biologia humana

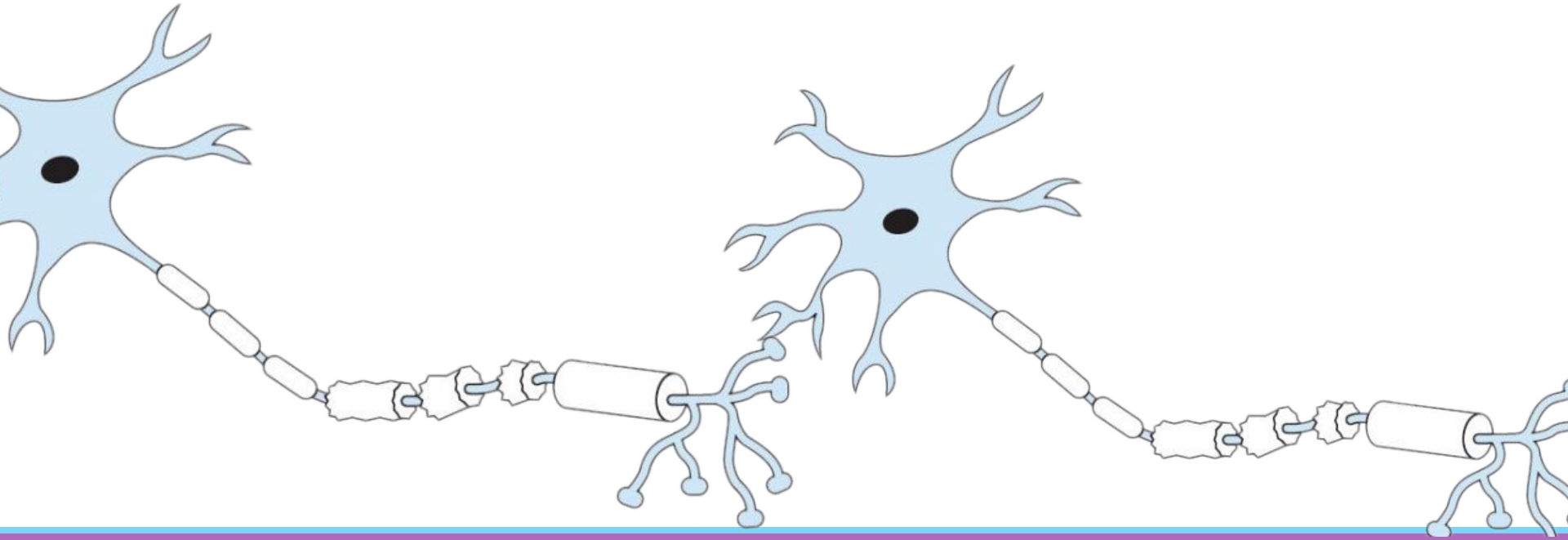


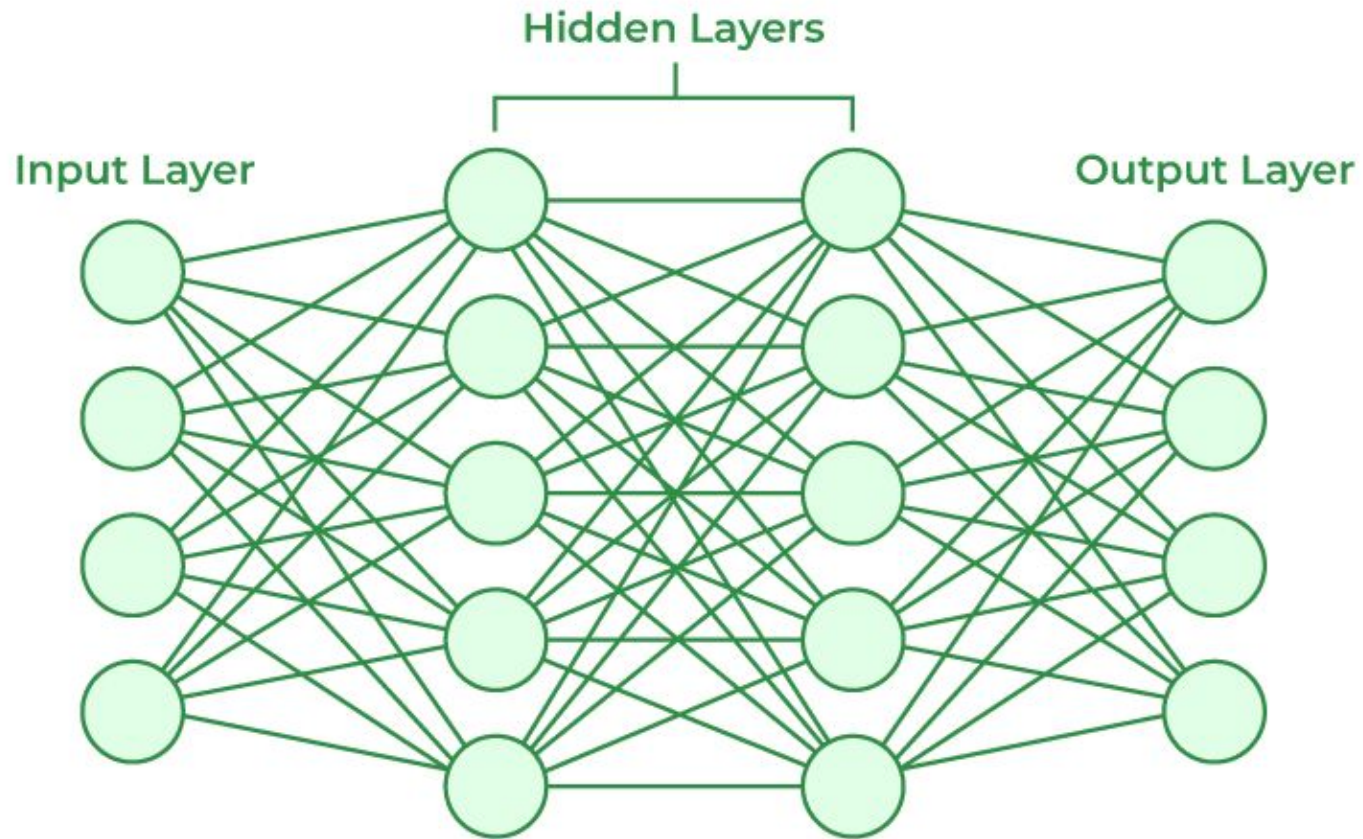
Impulso elétrico gerado
no corpo celular

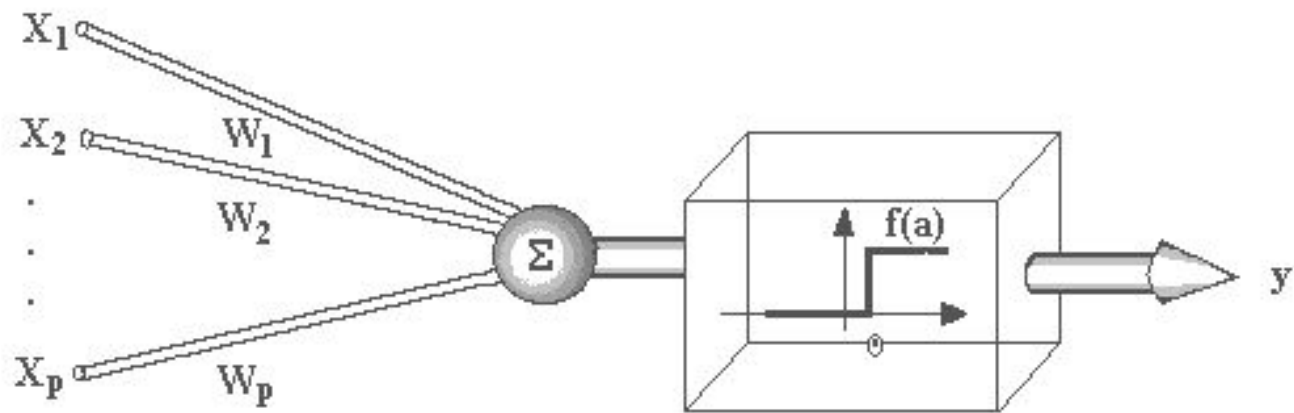
Impulso transmitido ao
longo do axônio

Os neurotransmissores
são liberados na sinapse

Na sinapse, os
neurotransmissores
podem ativar ou inibir os
receptores







Multiplicando pelo peso

$$x.w = (x_1 \times w_1) + (x_2 \times w_2) + \cdots + (x_n \times w_n)$$

Somando o Viés

$$z = x.w + b$$



Propriedades fundamentais de uma função ativação:

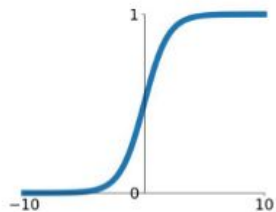
- Não-linearidade → introduzir não-linearidade na rede.
- Diferenciabilidade: → essencial para a retropropagação
depende do cálculo de gradientes da
função de perda
- Limitação ou estabilidade: → a estabilizar o treinamento,
impedindo que os valores de
ativação se tornem muito grandes
ou muito pequenos



Activation Functions

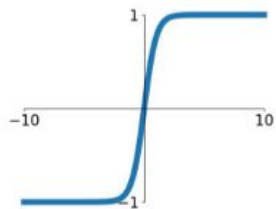
Sigmoid

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



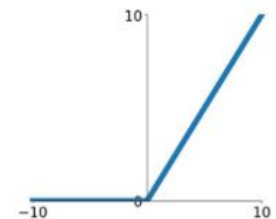
tanh

$$\tanh(x)$$



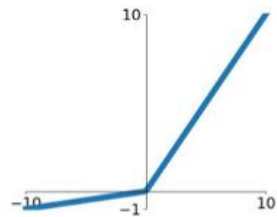
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

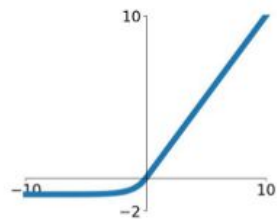


Maxout

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

ELU

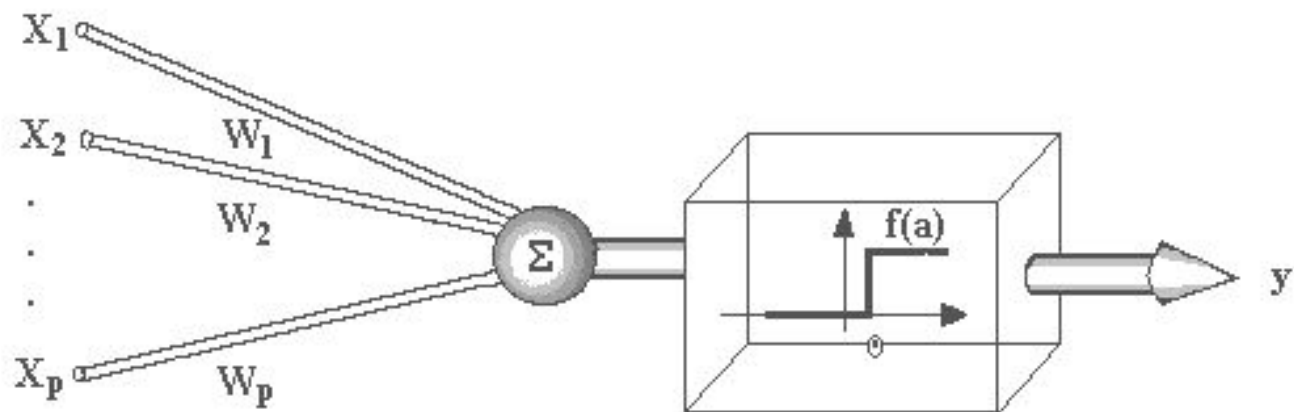
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$

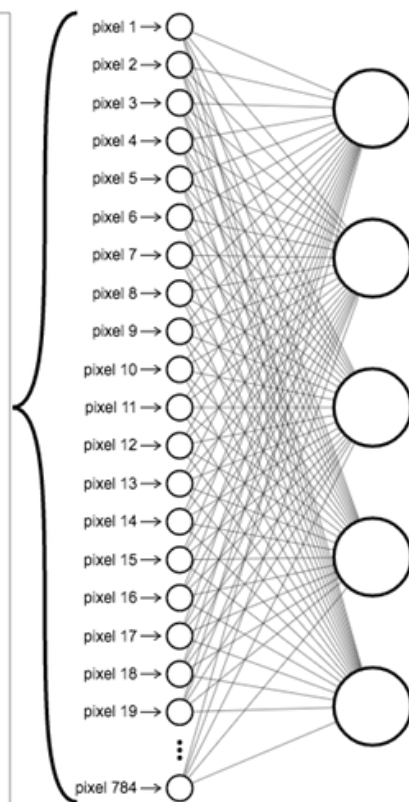
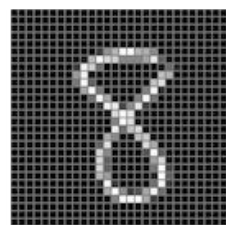


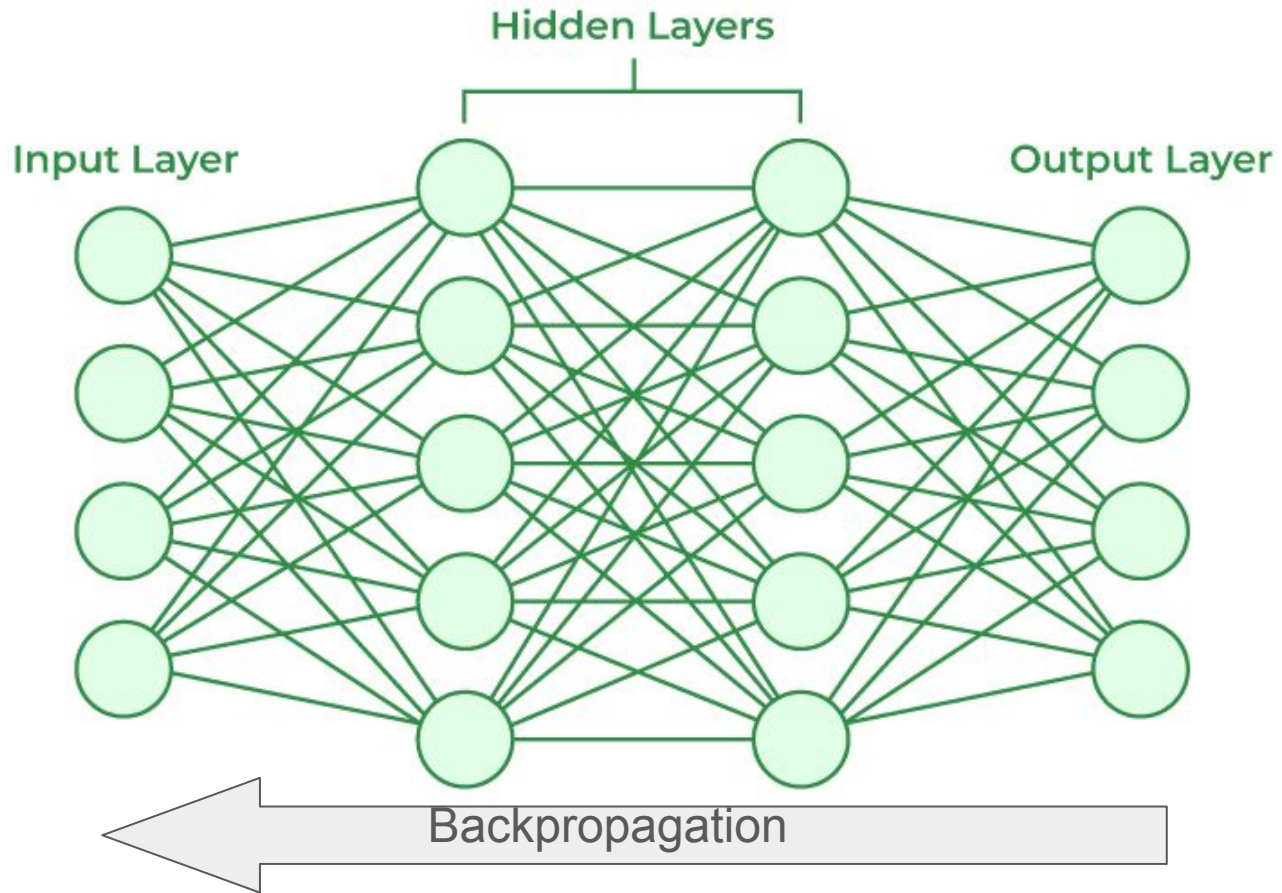
Aplicando o valor na função ativação

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

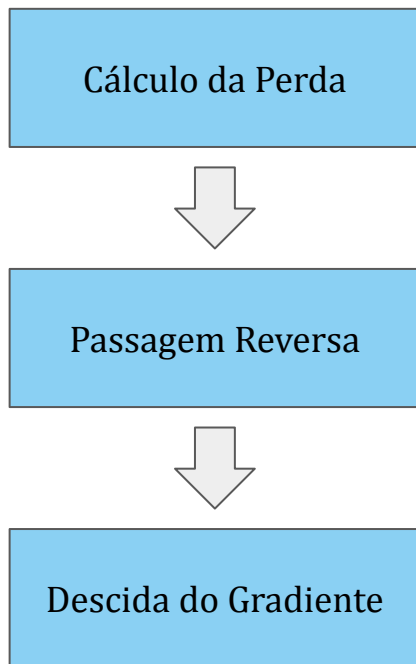








Backpropagation

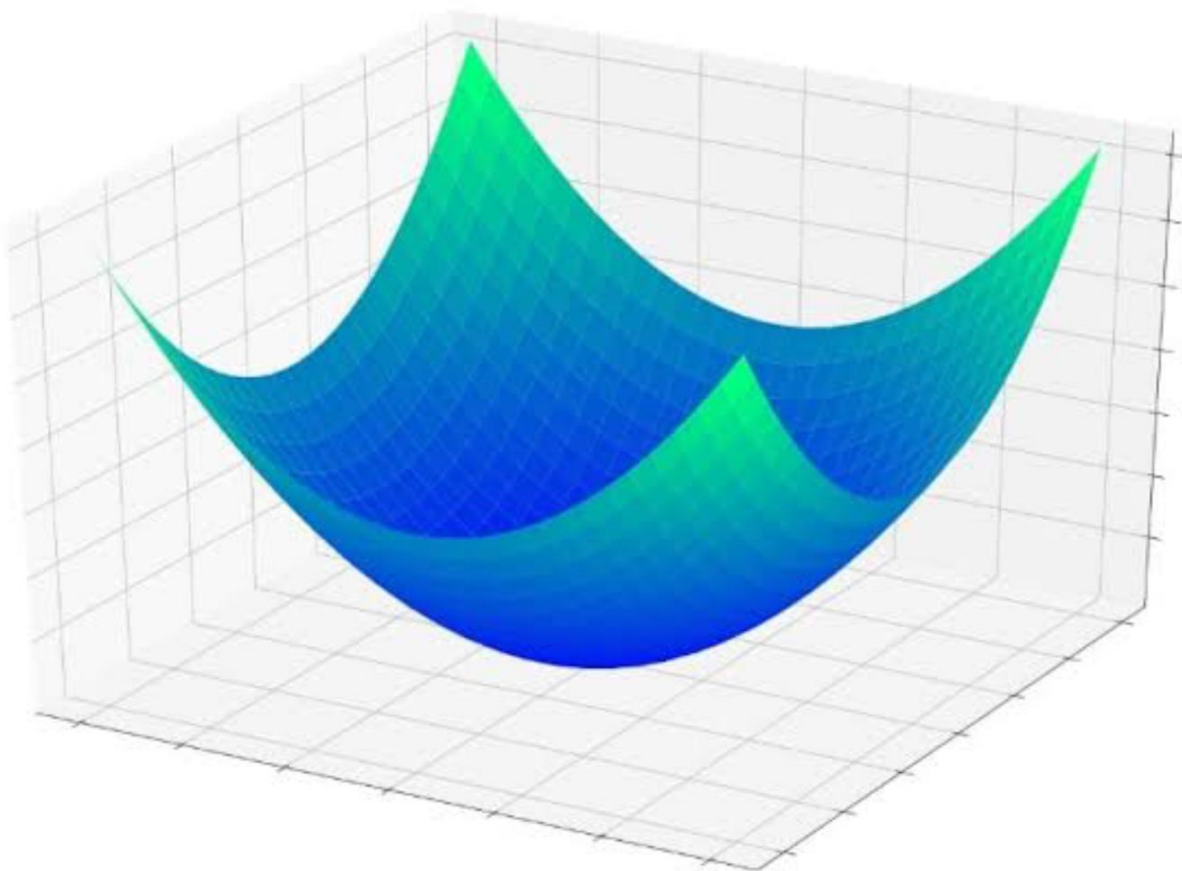


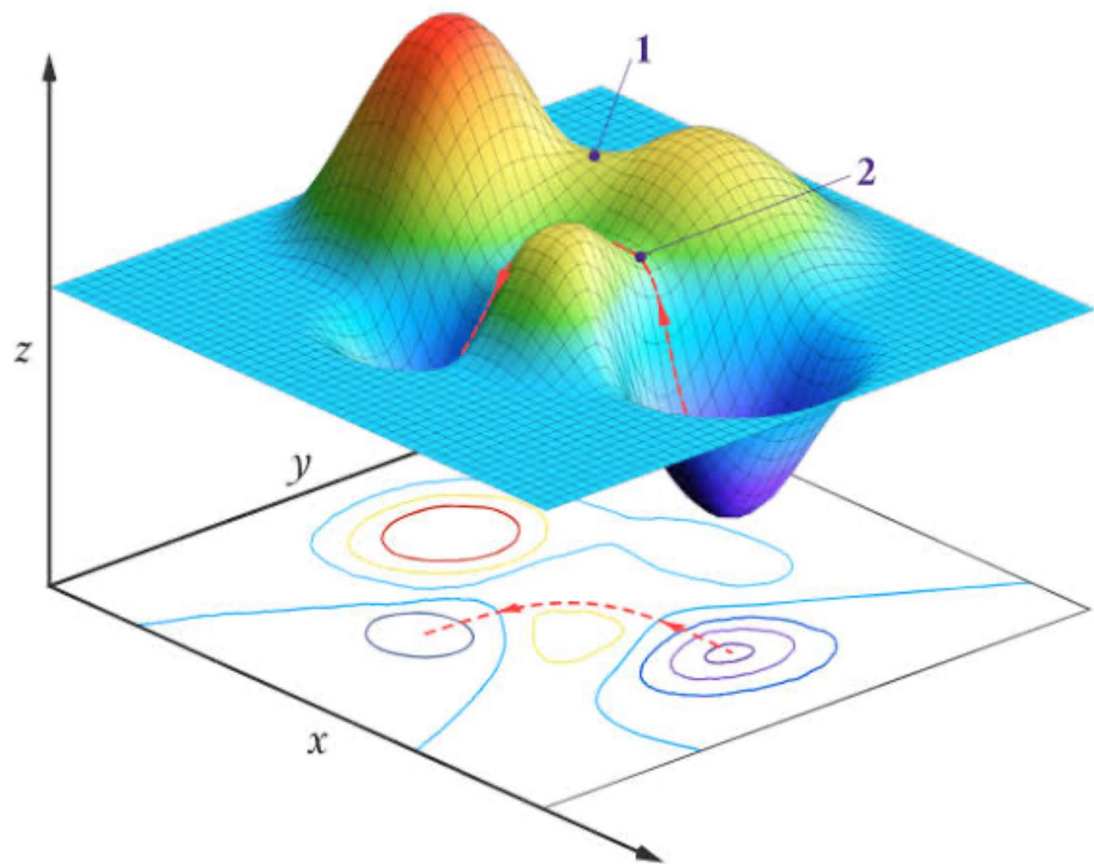
Após a passagem direta, a saída da rede neural é comparada à verdade absoluta (os valores alvo reais) usando uma função de perda pré-definida.

Na passagem reversa, o algoritmo calcula o gradiente da função de perda em relação a cada peso na rede usando a regra da cadeia do cálculo.

Uma vez que os gradientes são calculados, o algoritmo atualiza os pesos da rede na direção oposta do gradiente para minimizar a função de perda. Visando encontrar os pesos ideais







Modos de treinamento

Em batch: a atualização da rede (backpropagation) ocorre depois de passarmos **todos os exemplos** (feed-forward) de treinamento.

- Estimativa precisa dos vetores do gradiente.
- Tende a cair em mínimos locais



Modos de treinamento

Online ou estocásticos: Atualiza a cada exemplo apresentado. Ou seja, busca-se minimizar o erro instantâneo.

- Melhor para quando há redundância.
- Menos suscetível a mínimos locais.



Modos de treinamento

Mini-batch:a cada época.

- Embaralha os exemplos de treinamento.
- Faz o feedforward com uma pequena parte dos exemplos
- Acumula o erro na saída nesse batch, só depois faz o backpropagation
- Faz sequencialmente para os próximos batches
- Retorna para o início



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



Treinamento

As redes neurais devem ser capazes de reconhecer padrões a partir das entradas, sendo assim há a necessidade de modificar adequadamente os pesos e o viés de cada neurônio.

Para esse ajuste de pesos, utiliza-se dados já obtidos. Há 3 paradigmas relacionados ao aprendizado.

Supervisionado:

A partir de uma base de dados com resultados esperados, tenta-se diminuir a taxa de erro da rede.

Não Supervisionado:

O dados obtidos não têm resultado esperado, a rede deve tentar agrupar dados parecidos.

Semi-Supervisionado:

Apenas parte da amostra tem rótulos, as não rotuladas devem ser classificadas pela sua proximidade com as já rotuladas.



Aprendizado por correção de erro - Supervisionado

Regra delta ou Gradiente Descendente

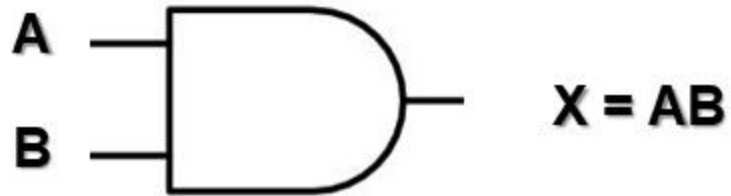
$$w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n)$$

$$\Delta w_{kj}(n) = \eta e_k(n) x_j(n)$$

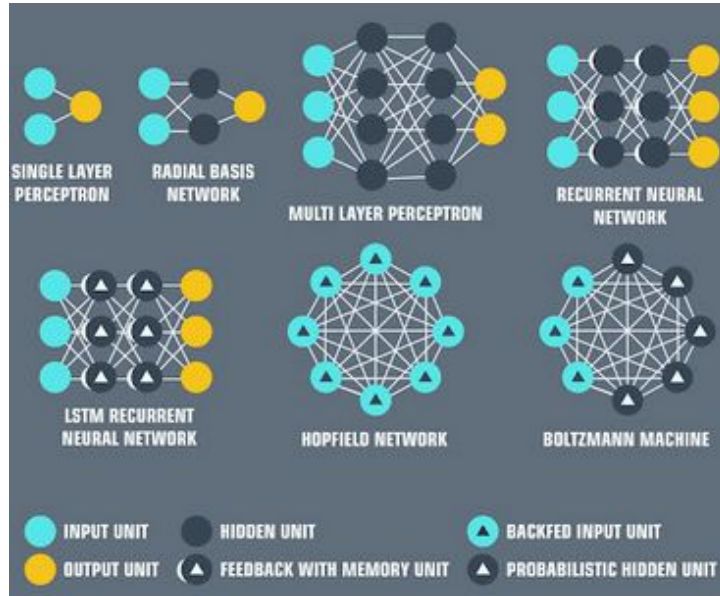


Perceptron - Simulação

Porta AND



Tipos de Redes Neurais



Multilayer Perceptron(MLP)

Supera as limitações práticas do Perceptron

- O modelo de cada neurônio inclui uma função de ativação não linear e diferenciável.
- Contém uma ou mais camadas escondidas/ocultas entre a camada de entrada e a camada de saída
- A rede possui alto grau de conectividade



Multilayer Perceptron(MLP)

Aprendizado pelo método de Backpropagation.

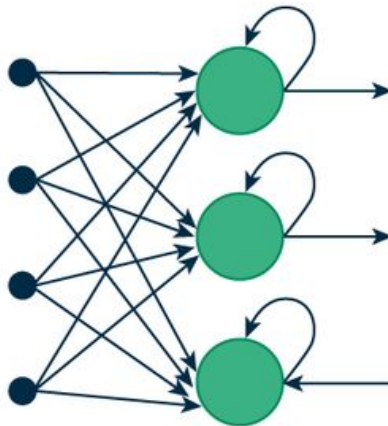
Forward Phase: pesos fixos e o sinal é propagado através da rede, camada por camada, até a saída.

Mudanças só ocorrem nos potenciais de ativação e nas saídas.



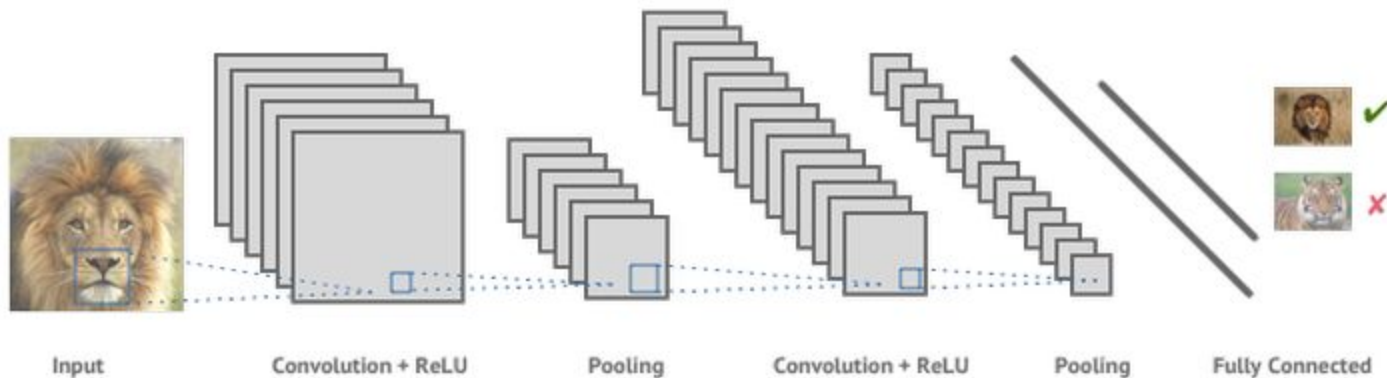
Rede Neural Recorrente

Uma rede neural recorrente (RNN) é um tipo de rede neural artificial que usa dados sequenciais ou dados de séries temporais. Esses algoritmos de aprendizado profundo são comumente usados para problemas ordinais ou temporais, como tradução de idiomas, processamento de linguagem natural (nlp), reconhecimento de fala e legenda de imagem;



Redes Neurais Convolucionais

As redes neurais convolucionais (CNNs) são especialmente projetadas para lidar com dados de entrada que possuem uma estrutura espacial, como imagens.



Referências Bibliográficas

Série de Aulas do professor Diego Silva sobre Aprendizado de Máquina -

<https://www.youtube.com/playlist?list=PL-K8PmZILhr0w0FH1gZrknk9CLRqXKo-Z>

3Blue1Brown - Neural Networks

https://www.youtube.com/watch?v=aircAruvnKk&list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi

MIT OpenCourseWareeu queria

<https://www.youtube.com/watch?v=uXt8qF2Zzfo>

