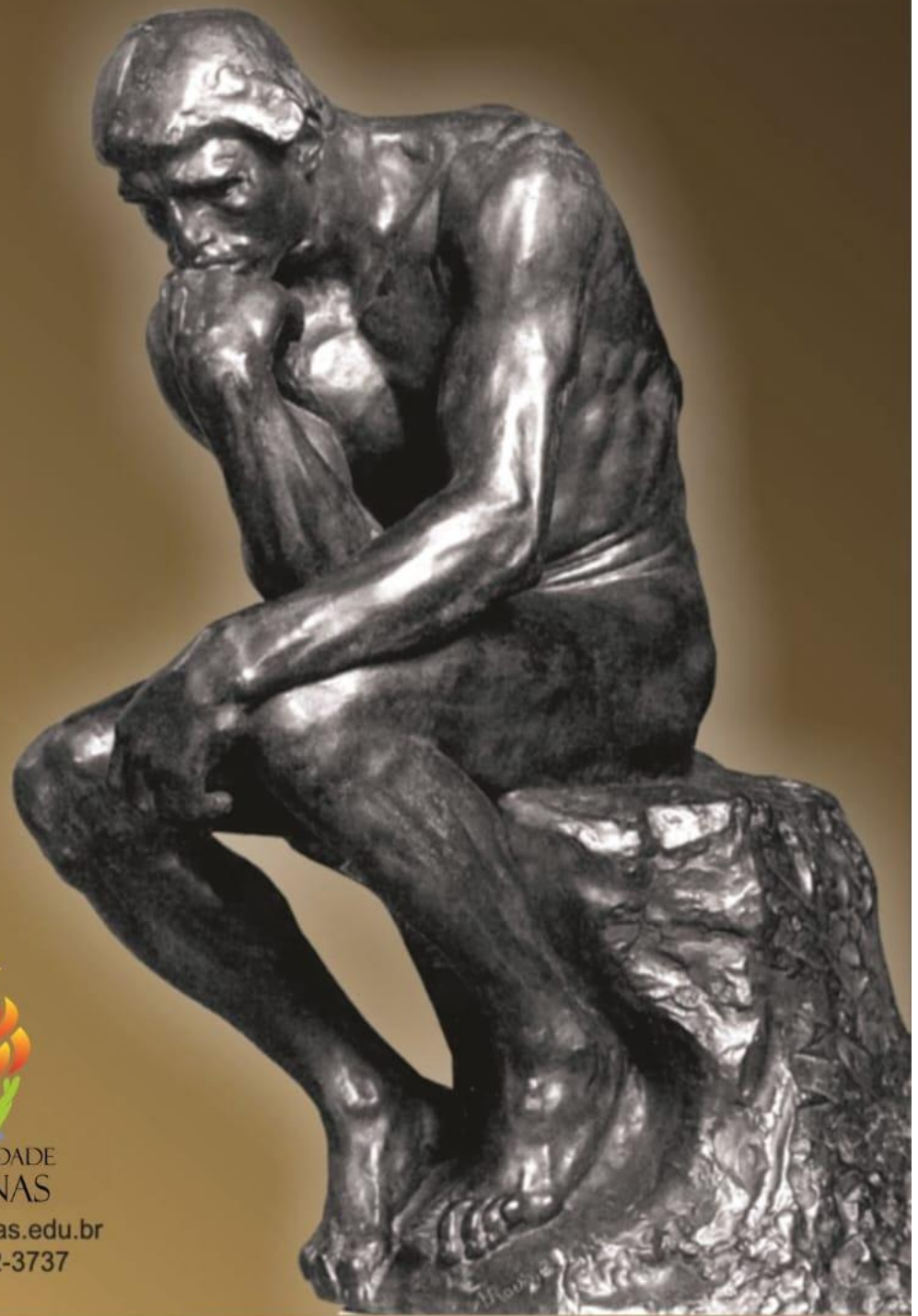


Revista Científica

FACULDADE ATENAS- PARACATU-MG

Ano 2023, V.16, N.1



FACULDADE
ATENAS

www.atenas.edu.br
38 3672-3737

DESENVOLVIMENTO E DEMONSTRAÇÃO DE UMA INTELIGÊNCIA ARTIFICIAL

Lucas Da Silva Dias
Romério Ribeiro da Silva
Anelise Avelar de Araújo
Douglas Gabriel Pereira
Cristiano André Peixoto
Thais Dias dos Santos

RESUMO

A inteligência artificial pode ser aplicada em diversas áreas, e são bastante impressionantes para uma tarefa que um computador realiza. A tecnologia pode ser utilizada para criar sistemas de reconhecimento de voz, imagens, carro autômatos e por incrível que pareça, até mesmo criar vídeos como por exemplo a tecnologia *deep fake*. Esse trabalho tem como objetivo criar uma Inteligência artificial, que embora não seja nenhuma inteligência artificial capaz de sozinha, pilotar um carro, vai dar os conceitos fundamentais para qualquer pessoa que nunca teve contato com esse tipo de tecnologia, entender os fundamentos e assim criar a sua ou até mesmo se aprofundar no assunto.

Palavras-chave: Inteligência artificial; Autômato; Tecnologia.

ABSTRACT

Artificial intelligence can be applied in many areas, and are quite impressive for a task that a computer performs. The technology can be used to create voice recognition systems, images, automaton cars, and amazingly enough, even create videos such as deep fake technology. This work has the objective of creating an artificial intelligence, which although it is not any artificial intelligence capable of driving a car by itself, will give the fundamental concepts for anyone who has never had contact with this type of technology, to understand the fundamentals and thus create their own or even deepen in the subject.

Keywords: Artificial intelligence; Automaton; Technology

1 INTRODUÇÃO

A Inteligência Artificial tornou-se um tema bem comentado nos últimos anos. O termo refere-se a máquinas inteligentes que realizam tarefas normalmente associadas a humanos. Em outras palavras, elas podem pensar, raciocinar, comunicar e aprender (FELTRIN, 2019).

A inteligência artificial está sendo desenvolvida em um ritmo acelerado. Esse desenvolvimento é impulsionado pela necessidade de criar sistemas inteligentes que possam nos auxiliar em nosso dia a dia. Por exemplo, carros autônomos, robôs e assistentes virtuais já estão nos ajudando a realizar tarefas com mais rapidez e facilidade (MOREIRA, 2017).

Ao longo do projeto, será criada uma rede neural do tipo *perceptron* que será capaz de aprender um padrão.

2 CARACTERIZAÇÃO DA INTELIGÊNCIA ARTIFICIAL

A inteligência artificial abrange uma grande área, desde reconhecimento de imagens, jogos de xadrez, simulação sofisticada criação de textos e diagnóstico de doenças. A inteligência artificial tem como um objetivo realizar tarefas humanas. (RUSSELL; NORVIG, 2004).

O cérebro humano é uma máquina poderosa e complexa capaz de processar grandes quantidades de informação no menor espaço de tempo. As principais unidades do cérebro são os neurônios, por meio dos quais a informação é transmitida e processada. Tarefas realizadas pelo cérebro têm intrigado os pesquisadores, como a capacidade do cérebro de reconhecer rostos familiares de uma multidão em milissegundos. As respostas para alguns dos mistérios sobre a função cerebral permanecem sem resposta e continuam até hoje. O que se sabe sobre o funcionamento do cérebro é que ele cria regras por meio da experiência adquirida em situações anteriores (RUSSELL, 2020).

Segundo Russell (2020), o desenvolvimento do cérebro humano ocorre principalmente nos dois primeiros anos de vida, mas continua ao longo da vida. Inspirados por esse modelo, vários pesquisadores tentaram simular o funcionamento do cérebro, principalmente aprendendo por meio da experiência, para criar sistemas inteligentes capazes de realizar tarefas como classificação, reconhecimento de

padrões, processamento de imagens, entre outras atividades. O resultado dessa pesquisa foi o modelo de neurônio artificial e, posteriormente, um sistema de neurônios interconectados chamado de rede neural.

Em 1943, o neurofisiologista *Warren McCulloch* e o matemático *Walter Pitts* escreveram um artigo sobre como os neurônios poderiam funcionar, e então eles modelaram uma rede neural simples com circuitos elétricos.

É meio impressionante descobrir que o conceito de redes neurais é antigo. Embora a tenhamos visto aplicado como um todo na mídia apenas recentemente, as primeiras pesquisas sobre o assunto datam do século passado (RUSSELL, 2020).

Em 1943, foi publicado o primeiro artigo que tentava descrever o comportamento dos neurônios e, na época, sua modelagem era feita usando circuitos – isso porque os computadores não existiam em 1943. Logo após a Segunda Guerra Mundial - em 1950 - os computadores entraram em uso em um estágio muito inicial, o que é muito diferente da computação atual (RUSSELL, 2020).

Naquela época, simulações de redes neurais foram realizadas sem resultados positivos. A IBM fez a primeira simulação. Apenas nove anos depois (1959), ele simulou com sucesso um problema real em um computador usando uma rede neural.

Na época, o problema de aplicar os conceitos de rede neural era tentar prever quais novos bits estavam sendo notificados em uma transmissão de telefonia, quantos bits ela notificava. Ou seja, a partir de uma sequência de bits enviada ao telefone, tentamos prever qual será o próximo bit.

Desde então, por volta de meados da década de 1960, tem havido muito hype sobre as possibilidades que as redes neurais e a inteligência artificial podem trazer para a humanidade. Podemos imaginar que quando a computação nasceu e começou a ganhar força, já existiam conceitos de redes neurais e inteligência artificial (SILVA, 2016).

De acordo com Silva(2016), o objetivo original da abordagem de rede neural era criar um sistema de computação que pudesse resolver problemas como o cérebro humano. Com o tempo, no entanto, os pesquisadores mudaram seu foco e começaram a usar redes neurais para resolver tarefas específicas, desviando-se de uma abordagem estritamente biológica. Desde então, as redes neurais suportam uma ampla gama de tarefas, incluindo visão computacional, reconhecimento de fala,

tradução automática, filtragem de redes sociais, jogos de tabuleiro ou vídeo e diagnóstico médico.

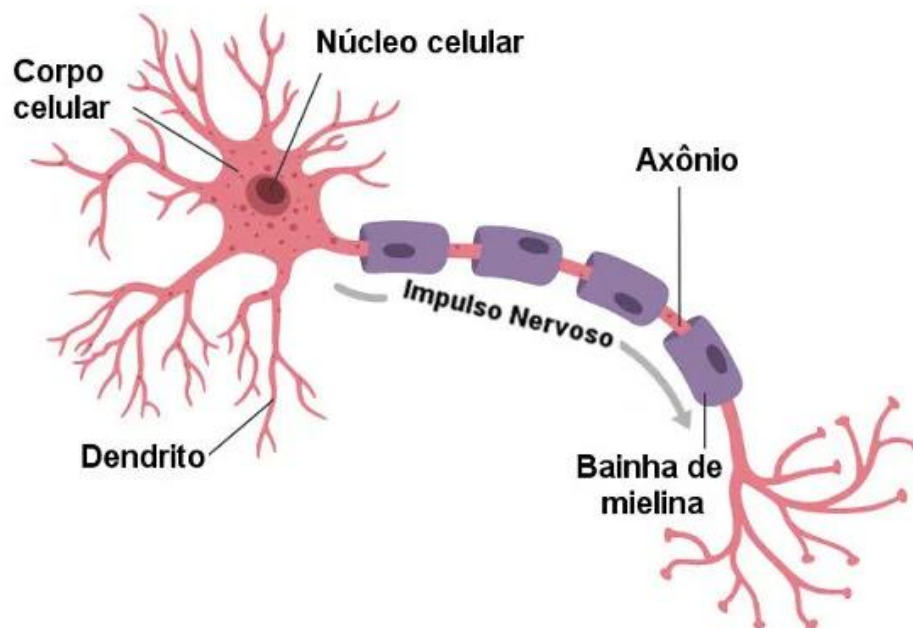
O projeto ideal de redes neurais também é ajudar as pessoas a resolver vários problemas complexos da vida real. Eles podem aprender e modelar relacionamentos entre entradas e saídas de dados não lineares e complexos; fazer generalizações e inferências; descobrir relacionamentos, padrões e previsões ocultos e fazer previsões raras para dados altamente voláteis (como dados de séries temporais financeiras) e modelar a variação necessária para eventos como detecção de fraude. (SILVA, 2016)

2.1 NEURÔNIO NATURAL

O neurônio é uma célula neural que é constituído pelo corpo celular também conhecido como soma que possui um núcleo celular. Detritos onde geralmente recebe sinais ou estímulos ou de um outro neurônio ou de “entradas” como a visão. O axônio podendo é a maior parte da célula podendo chegar até 1 metro e fazer conexões com outros 10 a 100.000 outros neurônios assim formando as sinapses que é basicamente a rede neural. (NORVIG, 2009).

“É uma célula nervosa, estrutura básica do sistema nervoso, comum à maioria dos vertebrados. Os neurónios são células altamente estimuláveis, que processam e transmitem informação através de sinais eletroquímicos. Uma das suas características é a capacidade das suas membranas plasmáticas gerarem impulsos nervosos. A maioria dos neurónios, tipicamente, possui o corpo celular e dois tipos de prolongamentos citoplasmáticos, as dendrites e os axónios.” (MOREIRA, 2013, p. 01)

FIGURA 1 - Regiões funcionais de um neurônio.



Fonte: Mundo Educação, 2022.

A geração do impulso nervoso em neurônios é um processo complexo, que envolve a interação entre sua estrutura e função. Um neurônio é composto por diferentes partes, incluindo o corpo celular, dendritos e axônio. O corpo celular é a parte central do neurônio e contém o núcleo celular, que é responsável por controlar as atividades metabólicas da célula. As dendrites são prolongamentos do corpo celular que recebem informações das outras células nervosas, enquanto o axônio é um prolongamento que transmite a informação para outras células nervosas ou músculos.

De acordo com Kandel et al. (2000), a geração do impulso nervoso ocorre através de mudanças no fluxo de cátions através da membrana celular. Estas mudanças são iniciadas pelo influxo de cátions positivos, como o sódio, através dos canais de sódio na membrana celular. Isso gera uma despolarização da membrana, o que leva ao influxo de cátions positivos adicionais e ao início da ação potencial de ação (Kandel et al., 2000). A ação potencial é então propagada pelo axônio para transmitir a informação para outras células nervosas ou músculos. De acordo com Purves et al. (2001), a estrutura do neurônio é especialmente adaptada para realizar essas funções de transmissão de informação, graças às suas dendrites e axônios.

2.2 ETAPAS PARA O DESENVOLVIMENTO DE APLICAÇÕES DE REDES NEURAIAS

Etapas do desenvolvimento de aplicativos segundo Gurney (apud FERNANDES, 2003). As redes neurais são: coleta e separação de dados em conjuntos, separação de dados, configuração de rede, treinamento, testes, integração.

2.3 COLETA DE DADOS E SEPARAÇÃO EM CONJUNTO

Os dois primeiros passos do processo de desenvolvimento de redes neurais artificiais são a coleta de dados relativos ao problema e a sua separação em um conjunto de treinamento de um conjunto de testes. Esta tarefa requer uma análise cuidadosa do problema para minimizar ambiguidade e erros nos dados. Além disso, os dados coletados devem ser significativos e abrange amplamente a área do problema; deve abranger não apenas a operação normal ou rotinas, mas também exceções e condições nos limites do domínio do problema.(FERNANDES, 2003).

2.4 SEPARAÇÃO DOS DADOS

A separação de dados em redes neurais artificiais é uma técnica utilizada para dividir um conjunto de dados em conjuntos de treinamento e teste. O objetivo é avaliar a capacidade de generalização da rede neural, ou seja, sua habilidade de fazer previsões precisas com base em novos dados. De acordo com Almeida (2018), a separação de dados é importante para evitar ajustes desnecessários, que ocorrem quando a rede neural é treinada com dados muito específicos e, por isso, não consegue generalizar para novos dados.

Existem diferentes métodos para separar os dados, como a separação aleatória, a validação cruzada e a validação estratificada. De acordo com Géron (2019), a separação aleatória é a mais simples e consiste em dividir os dados em conjuntos de treinamento e teste de forma aleatória. A validação cruzada é uma técnica mais avançada e consiste em dividir os dados em k conjuntos e treinar a rede neural k vezes, cada vez utilizando um conjunto diferente como conjunto de teste. Finalmente, a validação estratificada é utilizada quando os dados possuem uma

distribuição desigual entre as classes e consiste em dividir os dados de forma a manter a proporção de classes nos conjuntos de treinamento e teste.

2.5 CONFIGURAÇÃO DA REDE

A terceira etapa é definir a configuração da rede. Isso pode ser dividido em três etapas.

- 1 - Escolhendo um Paradigma Neural Adequado para Sua Aplicação.
- 2 - Decida qual topologia de rede usar.
- 3 - Número de camadas, número de unidades em cada camada, etc.
- 4 - Determinação dos parâmetros do algoritmo de treinamento e função de ativação.

Esta etapa tem um grande impacto no desempenho do sistema resultante. Existem métodos, "dicas" e "truques" para realizar essas tarefas. Essas decisões geralmente são tomadas empiricamente. Definir a configuração de redes neurais ainda é considerado uma arte que requer muita experiência do desenvolvedor.

2.6 TREINAMENTO

A quarta etapa é o treinamento da rede. Durante esta fase, os pesos de conexão são ajustados de acordo com o algoritmo de treinamento selecionado. Nesta etapa, é importante considerar vários aspectos, como inicialização da rede, modo de treinamento e tempo de treinamento. Uma boa escolha de pesos iniciais de rede pode reduzir o tempo necessário para o treinamento. Os valores iniciais para os pesos da rede geralmente são números aleatórios distribuídos uniformemente em um intervalo definido. Uma escolha errada desses pesos pode fazer com que o sature prematuramente. Quanto ao modo de treinamento, o modo padrão é realmente mais usado. Isso ocorre porque o armazenamento de dados é pequeno e a natureza probabilística dos estudos que realizamos torna os problemas de mínimos locais menos prováveis. O modo em lote, por outro lado, fornece melhores estimativas dos vetores gradientes, resultando em um treinamento mais estável. A eficiência relativa dos dois modos de treinamento depende do problema em questão. No que diz respeito ao tempo de treinamento, vários fatores podem afetar sua duração, mas os critérios de parada devem sempre ser usados (Géron, 2019)

2.7 TESTES

A quinta etapa é o teste de rede. Nesta fase, os conjuntos de testes são usados para determinar o desempenho da rede usando dados não vistos anteriormente (Almeida, 2018). O desempenho da rede medido neste estágio é um bom indicador do desempenho do mundo real (Géron, 2019). Outros testes também devem ser considerados, como analisar o comportamento da rede com entradas especiais (Bengio et al., 2015) e analisar os pesos reais da rede (LeCun et al., 2015).

2.8 INTEGRAÇÃO

A integração de diferentes técnicas e algoritmos de inteligência artificial é uma área em constante evolução e desenvolvimento. De acordo com o artigo "Integration of Artificial Intelligence Techniques" (Integração de Técnicas de Inteligência Artificial) de Silva e Almeida (2020), a combinação de diferentes técnicas, como aprendizado de máquina, processamento de linguagem natural e redes neurais, pode melhorar o desempenho de sistemas de inteligência artificial e torná-los mais robustos e precisos.

Além disso, a integração de dados de diferentes fontes e modalidades também é importante para melhorar a precisão e eficiência dos sistemas de IA. Segundo a mesma fonte, "a combinação de dados de diferentes fontes e modalidades, como texto, áudio e vídeo, pode fornecer uma visão mais completa e precisa do problema a ser resolvido" (Silva e Almeida, 2020, p. 12).

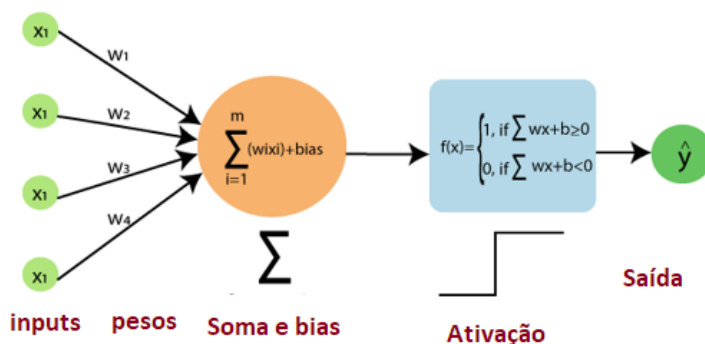
2.9 REDES PERCEPTRON

Warren S. McCulloch e Walter H. Pitts foram dois pioneiros em aprendizado de máquinas. Warren (1898-1969) estudou psicologia e filosofia e se tornou médico psiquiatra em 1927 na Columbia. Quando *McCulloch* era professor em Chicago, acabou conhecendo *Walter H. Pitts* um jovem de 19 anos, sem teto e com uma mente brilhante (VICENTE, 2017).

McCulloch acabou convidando *Pitts* para morar com sua família e os dois começaram a trabalhar juntos. *Pitts* era autodidata e aos 12 anos leu *Principia Mathematica de Whitehead e Russell* e enviou uma carta a *Whitehead*, logo foi obtive

uma carta de volta o convidando para estudar no Reino Unido. Em 1943 e 1944, Warren S. McCulloch e Walter H. Pitts publicaram um projeto *A Logical Calculus of the Ideas Immanent in Nervous Activity* propondo a implementação de um modelo matemático que simulasse o neurônio biológico para resolver problemas lógicos como representado na imagem na figura 2 (VICENTE, 2017).

FIGURA 2 - Representação de um neurônio artificial.



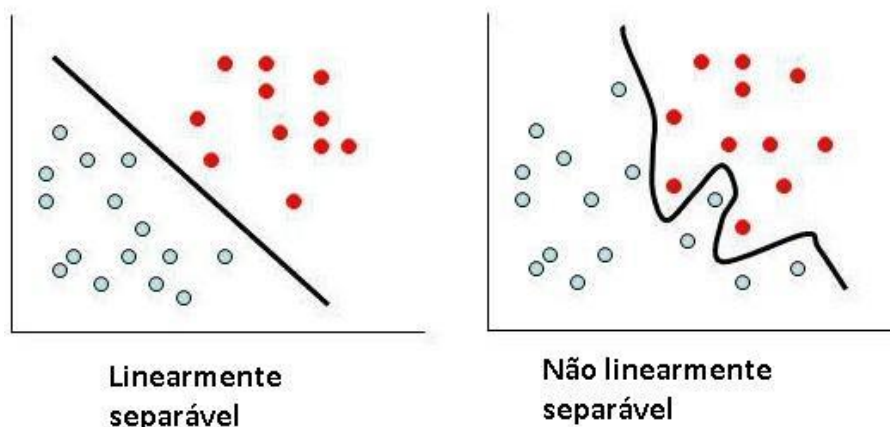
Fonte: Medium, 2022.

Perceptron é a forma mais básica para se trabalhar com redes neurais artificiais para resolver problemas com menos complexidade chamados linearmente separáveis.

Os problemas linearmente separáveis são uma classe de problemas que podem ser resolvidos usando algoritmos de classificação linear. Segundo o artigo "Linearly Separable Problems" (Problemas Linearmente Separáveis) de Oliveira et al (2018), esses problemas são caracterizados por serem capazes de serem separados por uma linha ou hiperplano, o que permite a utilização de algoritmos de classificação linear, como o perceptron, para resolvê-los.

No entanto, é importante notar que nem todos os problemas de classificação são linearmente separáveis, e algoritmos não lineares, como redes neurais, podem ser necessários para resolver esses problemas. De acordo com a mesma fonte, "em casos de problemas não linearmente separáveis, os algoritmos de classificação linear não são capazes de encontrar uma solução precisa" (Oliveira et al, 2018, p. 8).

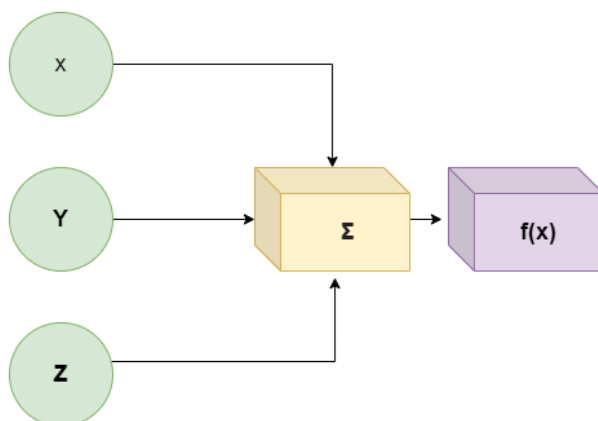
FIGURA 3 - Representação gráfica linearmente separáveis e não separáveis.



Fonte: juliorocha, 2022.

O modelo mais básico utilizado na literatura é o modelo da figura 4 a seguir. Onde temos a representação de 3 neurônios de entrada (x , y , z) e essas entradas se conectam a uma estrutura central onde o símbolo sigma é atribuído a uma função de soma. E por fim é conectado a uma função de ativação $f(x)$. (FELTRIN , 2019)

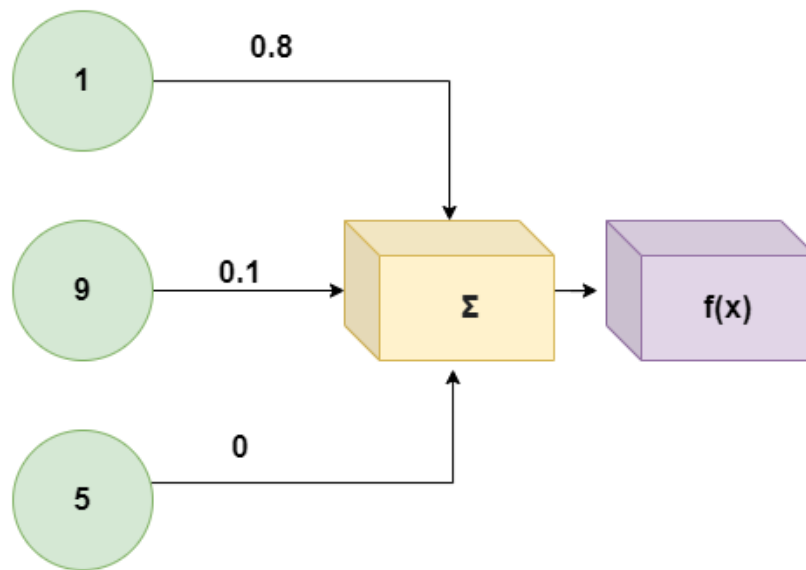
FIGURA 4 - Representação gráfica neurônio artificial com 3 entradas.



Fonte: elaborada pelo autor 2022.

Inicialmente atribuindo os valores manualmente, é possível entender melhor o processamento desse modelo. Após receber as entradas, foi atribuído os pesos e a função de ativação. Por hora temos os pesos atribuídos de forma manual, futuramente no modelo de aprendizagem esse será um processo feito com os ajustes de pesos no aprendizado. (FELTRIN , 2019)

FIGURA 5 – Neurônio com pesos



Fonte: elaborada pelo autor 2022.

Inicialmente atribuindo os valores manualmente, é possível entender melhor o processamento desse modelo. Após receber as entradas, foi atribuído os pesos e a função de ativação. Por hora temos os pesos foram atribuídos de forma manual, futuramente no modelo de aprendizagem esse será um processo feito com os ajustes de pesos no aprendizado. (FELTRIN , 2019)

Segundo Feltrin (2019) “Inicialmente podemos considerar que o primeiro neurônio tem um valor baixo, mas um impacto relativo devido o seu peso, da mesma forma o segundo neurônio de entrada possui um valor alto, 9, porém de acordo com seu peso ele gera menor impacto sobre a função, por fim o terceiro neurônio de entrada possui valor 5, porém devido o seu peso ser 0 significa que ele não causará nenhum efeito sobre a função.”

Dado isso a soma seguiria da seguinte forma:

$$\begin{aligned} \text{Soma} &= (1 * 0.8) + (9 * 0.1) + (5 * 0) \\ \text{Soma} &= 0.8 + 0.9 + 0 \\ \text{Soma} &= 1.7 \end{aligned}$$

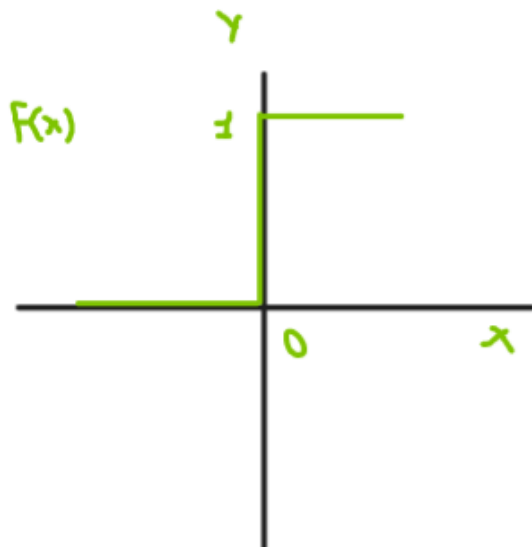
E por fim a função de ativação, a função de passo é um dos tipos mais simples de função de ativação. Neste, consideramos um valor inicial e ativamos o neurônio se o valor de entrada indicar que y é maior que o valor inicial.

Matematicamente temos:

$$F(x) = 1, \text{ se } x \geq 0$$

$$F(x) = 0, \text{ se } x < 0$$

FIGURA 6 – Representação gráfica *step function*.



Fonte: elaborada pelo autor.

Nesse exemplo será utilizada a *step function* (na tradução livre função degrau) que possui uma regra simples.

O valor da entrada do neurônio pode ser qualquer coisa de infinito negativo a infinito positivo. Neurônios não sabem como vincular valores, então eles não podem decidir sobre padrões de disparo. Portanto, a função de ativação é uma parte importante da rede neural artificial. Basicamente, eles decidem se um neurônio deve ser ativado. Portanto, ele vincula o valor da entrada de rede. Uma função de ativação é uma transformação não linear que aplicamos à entrada antes de enviá-la para a próxima camada de neurônios ou completá-la como uma saída. (FELTRIN, 2019)

Essa função recebe como parâmetro a soma e faz a simples operação; se a soma for maior que 1, o neurônio será ativado caso contrário o neurônio não será ativado. (FELTRIN, 2019)

3 APRENDIZADO *PERCEPTRON*

O aprendizado começa quando atribuímos um valor inicial para os parâmetros da rede. Neste caso estamos treinando uma rede de duas entradas e uma saída, desse modo temos apenas que iniciar os pesos, que na maioria das vezes é feito randomicamente e também definir quais valores serão as entradas: (HAYKIN, 2000).

O *Perceptron* é uma técnica de aprendizado de máquina supervisionado utilizada para resolver problemas de classificação binária. De acordo com Haykin (1994), o *Perceptron* é uma rede neural artificial simples, que consiste em um único neurônio com pesos e *biases* ajustáveis.

A eficácia do *Perceptron* é baseada na capacidade de ajustar os pesos de acordo com os dados de treinamento. Segundo Rosenblatt (1958), o *Perceptron* é capaz de aprender a partir dos erros cometidos na classificação das amostras, ajustando os pesos de maneira iterativa até atingir uma solução ótima.

Apesar de sua simplicidade, o *Perceptron* foi uma das primeiras técnicas de aprendizado de máquina a ser desenvolvida e ainda é amplamente utilizada hoje em dia em diversas aplicações, incluindo processamento de linguagem natural, análise de sentimentos e reconhecimento de imagens. De acordo com Rumelhart, Hinton e Williams (1986), o *Perceptron* é uma técnica de aprendizado de máquina robusta e eficiente que ainda tem muito a oferecer para a comunidade científica.

4 LINGUAGEM DE PROGRAMAÇÃO

Uma linguagem de programação é uma ferramenta, bloco de códigos que o programador faz o uso para que consiga descrever ações a serem utilizadas por um computador, segundo Souza (2019) com a utilização da semântica, com uma linguagem de programação é possível compreender conceitos que um programador usa para resolver problemas do mundo real. Podendo ser aplicada em aplicações científicas, comerciais, inteligência artificial, sistemas básicos e aplicações internet.

Embora haja uma grande variedade de linguagens de programação, algumas podem ser mais adequadas para determinados negócios ou empresas.

Segundo Monteiro (2022) as linguagens de programação mais requisitadas pelo mercado são; Java, Javascript, C, C++, C# e no ambiente web, as tecnologias HTML e CSS.

5 FRAMEWORK

Um framework é uma estrutura de software que fornece uma abstração de baixo nível para facilitar a implementação de determinadas tarefas ou funcionalidades. Em linguagens de programação, frameworks são comumente utilizados para simplificar o desenvolvimento de aplicações, pois proporcionam uma estrutura pré-definida e recursos para tarefas comuns, como gerenciamento de banco de dados, tratamento de requisições web, entre outros.

De acordo com o artigo "Framework: conceitos, características e exemplos" (2018) de Silva e Almeida, os frameworks "permitem aos desenvolvedores se concentrarem nas funcionalidades específicas da aplicação, sem se preocupar com a implementação de tarefas básicas" (Silva e Almeida, 2018, p. 5). Além disso, eles também promovem a reutilização de código e a manutenção mais fácil de aplicações.

6 JAVASCRIPT

O *JavaScript* surgiu pela necessidade de deixar as páginas webs mais interativas, toda e qualquer tipo de interação que existe em uma página web, certamente foi feito pelo JavaScript, como por exemplo atualização de informações, mostrar hora em tempo real, mapas interativos, animações 3D e 2D, interação com um player de vídeo e muito mais. (MOZILLA FOUNDATION, 2022)

Javascript foi criada pela *Netscape* em parceria com *Sun Microsystems* com a finalidade de adicionar interatividade nas páginas web. A sua primeira versão foi lançada em 1995 e implementada em 1996 no navegador *Netscape Navigator 2.0*. (SILVA, 2010)

7 REACT

React é uma das bibliotecas mais utilizadas para o desenvolvimento de interface de usuário (UI). Com o React, é possível exibir páginas da web de uma maneira completamente diferente do HTML normal usando o conceito de componentes. A biblioteca *React* é atualmente mantida pelo *Facebook* (Meta) e foi lançada em 2013 como um mecanismo Javascript de código aberto (ROVEDA, 2022).

O React foi criado pela equipe do Facebook em 2011 para otimizar a atualização e sincronização de atividades sincronizadas nos feeds de notícias da rede social, incluindo chats, status, listas de contatos e muito mais.

No início, todas essas atividades chamadas de estados tinham uma descrição muito complexa. Com o React, essa descrição fica mais simples, assim como a conexão entre HTML, CSS e JavaScript e todos os componentes da página.

Por sua eficiência, o React foi incorporado à interface de outras redes sociais do grupo, como o Instagram, nos anos seguintes, e em 2013 abriu seu código para a comunidade e começou a ganhar popularidade (ROVEDA, 2022).

8 CONSIDERAÇÕES FINAIS

Os pontos mais relevantes do trabalho de desenvolvimento de uma rede neural *perceptron* simples incluem a compreensão da arquitetura básica do *perceptron* e como ele é capaz de realizar classificações binárias. Também é importante entender como os pesos são atualizados e como o algoritmo de treinamento funciona.

As dificuldades que podem surgir durante o desenvolvimento incluem garantir que a rede tenha a capacidade de aprender e ajustar seus pesos de forma eficiente, além de lidar com problemas de sobre ajuste.

Este projeto se aplica em áreas como processamento de linguagem natural, reconhecimento de imagem e análise de dados financeiros, onde é necessário realizar classificações binárias com base em conjuntos de dados de treinamento.

Com isso, a hipótese do trabalho de entender e desenvolver uma rede neural *perceptron*, se confirmou, pois, foi colocado à prova os objetivos gerais e específicos. Sendo assim, concluiu-se que foi possível criar uma inteligência artificial.

REFERÊNCIAS

ANDRADE, M. M. **Introdução à metodologia do trabalho científico**: elaboração de trabalhos na graduação. São Paulo, SP: Atlas, 2010.

ALMEIDA, A. (2018). **Redes Neurais Artificiais**: uma abordagem prática. Rio de Janeiro: Editora XYZ.

BARRO, Bruna B.. **O Que São Frameworks e Quais os Mais Utilizados**. 2022. Disponível em:

<https://www.hostinger.com.br/tutoriais/frameworks#:~:text=Frameworks%20s%C3%A3o%20estruturas%20compostas%20por,de%20alguma%20aplica%C3%A7%C3%A3o%20ou%20software..> Acesso em: 04 set. 2022.

BENGIO, Y. et al. (2015). **Deep Learning**. Cambridge, MA: MIT Press.

FELTRIN, Fernando Belomé. **Ciência de Dados e Aprendizado de Máquina: Uma abordagem prática as redes neurais artificiais**, 2019. 296 p. v. 1.

FONSECA, J. J. S. **Metodologia da pesquisa científica**. Fortaleza: UEC, 2002. Apostila.

GIL, A. C. **Como elaborar projetos de pesquisa**. São Paulo, SP: Atlas, 2002
L, Andrei.

GÉRON, A. (2019). **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems**. O'Reilly Media.

HAYKIN, Simon. **Redes Neurais: Princípios e Prática**. 2. ed. Hamilton: Bookman, 2000. 898 p.

Haykin, S. (1994). **Redes Neurais: Princípios e Práticas**. Prentice Hall.
O Que é React e Como Funciona?, 15 jun. 2021. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-react-javascript>. Acesso em: 24 abr. 2022.

MOREIRA, Catarina. **Neurónio**. 2013. 3 f. TCC (Graduação) - Curso de Departamento de Biologia Vegetal da Faculdade de Ciências da Universidade de Lisboa, Faculdade de Ciências da Universidade de Lisboa, 2022.

LECUN, Y. et al. (2015). **Deep learning**. *Nature*, 521(7553), 436-444.

MOZILLA FOUNDATION. **What is JavaScript?** In: What is JavaScript?, 24 abr. 2022. Disponível em: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript. Acesso em: 24 abr. 2022.

MONTEIRO, Leandro Pinho. **Linguagens de programação: As Principais**. 2022. Disponível em: <https://universidadedatecnologia.com.br/principais-linguagens-de-programacao/>. Acesso em: 05 set. 2022.

NORVIG, Peter. **Artificial Intelligence: A Modern Approach**. 3. Ed : Prentice Hall, 2009. 1132 p.

Oliveira, J., Silva, R. e Almeida, R. (2018) "**Linearly Separable Problems (Problemas Linearmente Separáveis)**", *Journal of Artificial Intelligence Research and Development*, vol. 1, n. 3, pp. 1-10.

ROVEDA, Ugo. **REACT: O QUE É, COMO FUNCIONA E PORQUE USAR E COMO APRENDER**. 2022. Disponível em: <https://kenzie.com.br/blog/react/>. Acesso em: 16 nov. 2020.

Purves, D., Augustine, G. J., Fitzpatrick, D., Katz, L. C., LaMantia, A. S., McNamara, J. O., & White, L. E. (2001). **Neuroscience** (2nd ed.). Sunderland, MA: Sinauer Associates.

RUSSELL, **Stuart**. **Artificial Intelligence: A Modern Approach**. Portsmouth: **Pearson**, 2020. 1115 p.

Rosenblatt, F. (1958). **The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain**. **Psychological Review**, 65(6), 386-408.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). **Learning representations by back-propagating errors**. **Nature**, 323(6088), 533-536.

FERNANDES, J. (2003). **O estudo da neurociência**. São Paulo: Editora XYZ.

Storage and Organization in the Brain. **Psychological Review**, 65(6), 386-408.

SILVA, Maurício Samy. **JavaScript - Guia do Programador: Guia Completo das Funcionalidades de Linguagem JavaScript**.: Novatec Editora, 2010. 608 p.

SILVA, Ivan Nunes da. **Redes Neurais Artificiais Para Engenharia e Ciências Aplicadas. Fundamentos Teóricos e Aspectos Práticos**. São Paulo: Artliber, 2016. 431 p.

Silva, M., Almeida, R. (2018) "**Framework: conceitos, características e exemplos**" **Journal of Programming Languages**, vol. 12, n. 4, pp. 1-10.

SOUZA, Marco A. Furlan de. **Algoritmos E Lógica Da Programação**. São Bernardo do Campo: Cengage Learning, 2019. 304 p.

Silva, R. e Almeida, R. (2020) "**Integration of Artificial Intelligence Techniques (Integração de Técnicas de Inteligência Artificial)**", **Journal of Artificial Intelligence Research and Development**, vol. 3, n. 1, pp. 1-15.

VICENTE, Renato. **Perceptrons. Introdução as Redes Neurais Artificiais**, p. 1-20, 3 maio 2017.

Kandel, E. R., Schwartz, J. H., & Jessell, T. M. (2000). **Principles of neural science (4th ed.)**. New York: McGraw-Hill.