

## **APLICAÇÃO DA LÓGICA PARACONSISTENTE E REDES NEURAIS ARTIFICIAIS NO AUXÍLIO AO DIAGNÓSTICO DO CERATOCONO**

### **APPLICATION OF PARACONSISTENT LOGIC AND ARTIFICIAL NEURAL NETWORKS IN THE AID OF KERATOCONUS DIAGNOSIS**

**Daniel dos Reis Cavalcante Neto**

Tecnologia em Análise e Desenvolvimento de Sistemas

**Matheus Guermandi Ribeiro**

Tecnologia em Análise e Desenvolvimento de Sistemas

**Prof. Dra. Luciene Cavalcanti Rodrigues**

Tecnologia em Análise e Desenvolvimento de Sistemas  
(Orientadora)

*Tendo em vista a crescente evolução da tecnologia computacional e a forma como ela se faz presente em nosso cotidiano, observa-se sua grande influência na Medicina, em uma correlação das duas áreas. O presente trabalho apresenta o estudo e desenvolvimento de um algoritmo capaz de analisar exames oftalmológicos obtidos a partir de Tomografias de Coerência Óptica (OCT), analisando e distinguindo as córneas saudáveis das córneas enfermas. O desenvolvimento deste projeto se torna possível por meio do estudo da Inteligência Artificial e Redes Neurais Artificiais, para a classificação dos dados e da Lógica Paraconsistente, para a seleção das melhores características analisadas. Através de um conjunto de vetores de características oriundos das tomografias de 177 pacientes, um algoritmo de aprendizado de máquina foi desenvolvido em linguagem de programação Python para o reconhecimento de padrões das OCTs de amostras e classificação de novos exames entre patológicos (com ceratocone) e saudáveis (sem ceratocone).*

**Palavras-chave:** Tomografia de Coerência Óptica. Inteligência Artificial. Redes Neurais Artificiais. Lógica Paraconsistente. Ceratocone. Aprendizado de Máquina. Reconhecimento de Padrões.

*In view of the increasing evolution of computer technology and the way it is present in our daily lives, its great influence in medicine is observed, in a correlation of the two areas. This work presents the study and development of an algorithm capable of analyzing ophthalmologic examinations obtained from Optical Coherence Tomographies (OCT),*

*analyzing and distinguishing healthy corneas from the sick corneas. The development of this project becomes possible through the study of Artificial Intelligence and Artificial Neural Networks, for the classification of data and Paraconsistent Logic, for the selection of the best analyzed characteristics. Through a set of vectors of characteristics derived from the tomographies of 177 patients, it is proposed the development of a machine learning algorithm in Python programming language for the recognition of patterns of ocs of samples and classification of new tests between pathological (with keratoconus) and healthy (without keratoconus).*

**Keywords:** *Optical Coherence Tomography. Artificial Intelligence. Artificial Neural Networks. Paraconsistent Logic. Keratoconus. Machine Learning. Pattern Recognition.*

## 1 INTRODUÇÃO

Nas últimas décadas, o mundo tem assistido a uma das maiores revoluções da sua história: a computação, que possibilitou o surgimento da rede mundial de computadores (a *internet*). Esses acontecimentos trouxeram novos recursos e propuseram grandes facilidades à humanidade, alterando drasticamente o modo e a qualidade de vida da população.

Atualmente, conta-se com a alta disponibilidade de recursos tecnológicos, que estão intensamente presentes no dia a dia, desde dispositivos vestíveis, eletrodomésticos automatizados e assistentes domésticos, a veículos autônomos controlados por algoritmos inteligentes, resultantes dos crescentes estudos na área de Inteligência Artificial (IA).

A revolução das máquinas trouxe grandes avanços às diversas áreas do conhecimento humano e, dentre tantos benefícios, contribuiu diretamente com a medicina e a forma como ela é exercida:

Ao mesmo tempo, houve um grande avanço nas tecnologias médicas voltadas ao diagnóstico pela imagem: o ecocardiograma, que, alguns acham, tornará obsoleto o estetoscópio, o ultrassom, que substitui, dizem, a palpação e a percussão do abdome, a ressonância, que substitui os exames de radiologia contrastada, e o PET, que permite verificar as áreas envolvidas em várias atividades cerebrais, vieram substituir o exame clínico dos pacientes e a relação médico-paciente. Hoje, faz-se diagnóstico pelos exames e não pelo raciocínio e julgamento clínico. Para que avaliar a expansão torácica, auscultar, sentir o frêmito toracovocal ou fazer uma percussão para diagnosticar uma pneumonite, se um raio X simples de tórax pode fazer o diagnóstico? (LOBO, 2018, p. 3)

Uma das maiores influências da computação na medicina deu-se por meio da IA, que, segundo Russel e Norvig (2013), é uma área de estudo universal, relevante para qualquer campo

de pesquisa. Ainda segundo os autores, ela envolve vários subcampos, desde soluções mais generalistas até atividades mais específicas, como o diagnóstico de doenças, um dos tópicos de pesquisa deste trabalho.

Definir exatamente o que é Inteligência Artificial é uma tarefa complexa. Vários autores propõem significados diferentes, desde afirmações mais técnicas, que associam computação à neurociência, a outras que pendem para o estudo da psicologia, economia e até da filosofia. Devido ao escopo técnico e prático deste trabalho, convém considerar a definição de Lobo (2018, p. 4) para Inteligência Artificial: “Inteligência artificial (IA) é um ramo da ciência da computação que, usando algoritmos definidos por especialistas, é capaz de reconhecer um problema ou uma tarefa a ser realizada, analisar dados e tomar decisões, simulando a capacidade humana”.

Este trabalho foi desenvolvido a partir de exames oftalmológicos de 177 pacientes não identificados, fornecidos pelo Prof. Dr. Gildásio Castello de Almeida Jr, da Faculdade de Medicina de São José do Rio Preto (FAMERP). Dentre os exames investigados, 34 são de pacientes com ceratocone e 143 de pacientes saudáveis.

Inicialmente, uma extração de dados manual é realizada sobre os exames oftalmológicos, obtendo os dados dos resultados em imagem para planilhas eletrônicas, a fim de se coletar dados sobre as espessuras da córnea e do epitélio ocular. Posteriormente, por meio do uso da Lógica Paraconsistente, as melhores características desses vetores são encontradas com o objetivo de otimizar o desempenho do algoritmo classificador, que terá como entrada apenas as características mais relevantes.

## **1.1 Objetivos**

Este trabalho tem por objetivo, de forma geral, o desenvolvimento de um algoritmo classificador capaz de analisar Tomografias de Coerência Óptica (OCT<sup>1</sup>) e distinguir córneas

---

<sup>1</sup> Tomografia de Coerência Óptica (OCT) é um exame oftalmológico não invasivo - sem contato físico com o olho do paciente - que produz imagens da estrutura ocular, com alta definição e tridimensionais. As imagens são produzidas a partir de um feixe de luz com espectro próximo ao infravermelho.

saudáveis, pelo seu mapa de epitélio, daquelas com ceratocone, enfermidade ocular que altera a curvatura da córnea afetando a capacidade visual do paciente.

Um classificador é um algoritmo dito “inteligente”, baseado nos estudos de Inteligência Artificial e Redes Neurais Artificiais e Aprendizado de Máquina, capaz de reconhecer padrões a partir de um “treinamento” prévio, no qual são fornecidas entradas conhecidas para que, posteriormente, ele seja capaz de determinar a que classe pertence um conjunto de dados com saída desconhecida (SILVA; SPATTI; FLAUZINO, 2019). As entradas propostas são os vetores de características obtidos dos exames oftalmológicos, que podem pertencer a duas classes: patológicos ou não patológicos.

A utilização do aparato tecnológico visa a auxiliar os profissionais da medicina oftalmológica no diagnóstico do ceratocone. O uso da Inteligência Artificial nessa situação não tem por objetivo substituir o médico oftalmologista, mas sim auxiliá-lo no exercício da profissão, de forma a tornar a identificação da patologia mais rápida e precisa, propiciando a execução de um tratamento mais eficaz.

Os objetivos específicos deste projeto são os seguintes:

- Estudo da patologia ceratocone.
- Estudo e aplicação da Inteligência Artificial e Redes Neurais Artificiais para o desenvolvimento do algoritmo classificador.
- Estudo e aplicação da Lógica Paraconsistente para a seleção das melhores características produzidas pela Tomografia de Coerência Óptica.
- Desenvolvimento e treinamento do algoritmo classificador.

## **1.2 Justificativa**

Para avaliar a relevância e viabilidade deste projeto, é necessário levar em consideração alguns aspectos, tal como a importância da computação na área da medicina. Sobre isso, Lobo (2018, p. 1) discorre:

Ao mesmo tempo em que se discutem problemas na relação médico-paciente e a deficiência do exame clínico na atenção médica, que torna o diagnóstico clínico mais dependente de exames complementares, enfatiza-se cada vez mais a importância do computador em medicina e na saúde pública. Isto se dá seja pela adoção de sistemas de apoio à decisão clínica, seja pelo uso integrado de novas tecnologias, incluindo as tecnologias vestíveis/corporais (*wearable devices*), seja pelo armazenamento de grandes volumes de dados de saúde de pacientes e da população.

A utilização da IA vem sendo muito requisitada para solucionar problemas de diferentes níveis de complexidade:

O mundo moderno está cheio de incertezas e contradições. À medida que a inteligência artificial (IA) avança, as máquinas são frequentemente esperadas para imitar o cérebro humano e, consequentemente, enfrentar os conflitos associados a essa tarefa. (GUIDO, 2019, p. 1, tradução nossa)<sup>2</sup>

Uma das áreas em que as máquinas são requisitadas para imitar o cérebro humano e enfrentar os conflitos e desafios associados é a medicina, na qual, segundo Lobo (2018), 32% dos erros médicos constatados nos Estados Unidos da América são provenientes da relação ruim entre médico e paciente, de um exame clínico ineficiente ou da falha na análise do resultado de exames, o que evidencia a necessidade de novos meios para auxiliar esse processo. Ainda segundo Lobo:

Inteligência Artificial em medicina é o uso de computadores que, analisando um grande volume de dados e seguindo algoritmos definidos por especialistas na matéria, são capazes de propor soluções para problemas médicos. (LOBO, 2018, p. 3).

Além da análise técnica, é importante observar como este trabalho atinge a sociedade. O ceratocone não afeta somente a visão do indivíduo, mas também a sua saúde mental, ou seja, torna-se também um problema psicológico e social (MOREIRA, L., ALCHIERI, BELFORT JUNIOR; MOREIRA, H. 2007).

---

<sup>2</sup> *Today's modern world is filled with uncertainties and contradictions. As artificial intelligence (AI) advances, machines are frequently expected to mimic the human brain and, consequently, face the conflicts associated with this task.*

Moreira, L., Alchieri, Belfort Junior. e Moreira, H. (2007) afirmam que a personalidade, além de ser um traço hereditário, também é formada pela interação genética da pessoa com o meio ambiente. Além disso, avaliam como as enfermidades e seus tratamentos afetam o comportamento e a qualidade de vida dos pacientes. Constataram, então, que o ceratocone piora o comportamento físico e mental dos seus portadores, tornando-os mais pessimistas, intuitivos, retraídos e inseguros.

O estudo e desenvolvimento deste trabalho justifica-se pela alta relevância da utilização e estudo de Inteligência Artificial e Redes Neurais Artificiais, pelo impacto positivo no exercício da medicina e pelos benefícios que o diagnóstico mais preciso do ceratocone propõe aos pacientes enfermos, que poderão ser tratados antecipadamente e de forma mais satisfatória.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 Ceratocone Subclínico

Com a finalidade de possibilitar uma melhor compreensão da patologia que será estudada, deve-se assimilar como a anatomia do olho humano é constituída e quais as funções de cada subdivisão.

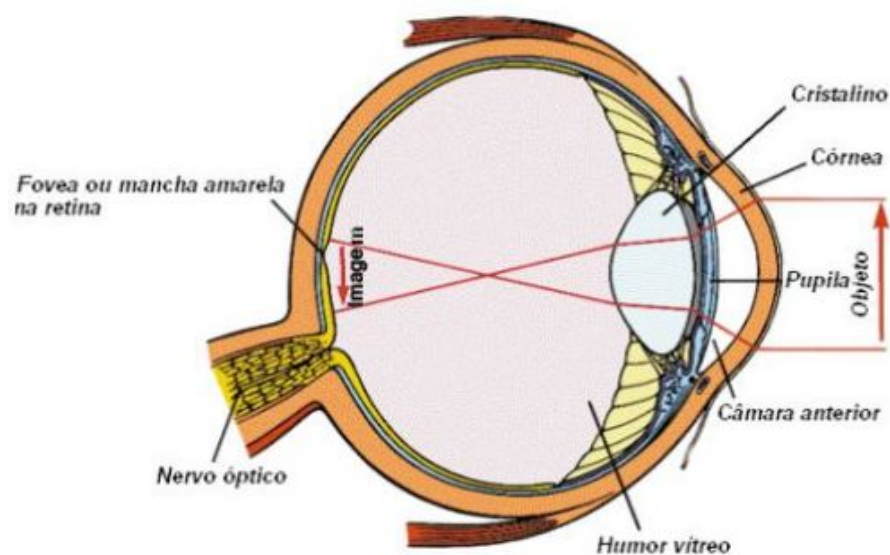


Figura 1 - Anatomia do olho humano  
Fonte: (RAMOS, 2006, p. 5)

#### 2.1.1 Anatomia do Olho

De acordo com Ramos (2006), a ação de olhar, ver ou enxergar decorre do órgão nomeado como *olho*. Ele é constituído por um grupo complexo de elementos que atuam em conjunto. Conforme a Figura 1, cada parte presente no órgão possui uma função a desempenhar, sendo elas especificadas a seguir:

- Córnea: camadas transparentes que proporcionam proteção ao órgão. Tem como função juntamente com o cristalino focar a luz através da pupila para a retina, como se fosse uma lente fixa (RAMOS, 2006).
- Íris: porção visível e colorida do olho. Possibilita que o órgão receba mais ou menos luz de acordo com a luminosidade do ambiente, utilizando músculos para aumentar ou diminuir a pupila (RAMOS, 2006).
- Pupila: orifício localizado na parte central do órgão que possui como principal função regular a quantidade de luz que alcança o cristalino (RAMOS, 2006).
- Cristalino: funciona como uma lente e participa dos meios de refração do olho. Dispõe da capacidade conhecida como “acomodação”, capaz de aumentar o grau para focalização das imagens de perto (RAMOS, 2006).
- Retina: membrana do segmento posterior do olho, cuja principal função é transmutar o estímulo luminoso em estímulo nervoso e submeter ao cérebro, para que assim seja interpretada como uma imagem (RAMOS, 2006).
- Nervo óptico: responsável por transmitir ao centro de processamento do cérebro os impulsos elétricos do olho para uma correspondente compreensão da imagem (RAMOS, 2006).
- Esclera: também conhecida como esclerótica, é a parte branca do olho que é conectada à córnea. É constituída por uma capa externa, fibrosa, branca e rígida que envolve o olho e dá forma ao globo ocular (RAMOS, 2006).

### **2.1.2 Patologia Ceratocone**

Benedict Duddell foi quem realizou uma descrição inicial do ceratocone. Essa descoberta se deu enquanto analisava as condições oculares de um jovem há mais de três séculos. Nesse contexto, ocorre que o ceratocone já foi confundido com várias outras enfermidades. Tão somente em 1854, no principal tratado sobre córnea cônica, escrito por John Nottingham, foi



determinado claramente o ceratocone e suas associações (MCGHEE, 2008; GRZYBOWSKI; MCGHEE, 2013).

O termo ceratocone é derivado de palavras gregas, conforme “*kerato*” significa “córnea” e “*conus*”, “cone” (WHEELER; HAUSER; AFSHARI; ALLINGHAM, 2013).

O ceratocone se caracteriza por ser uma anomalia habitual que acomete a córnea de forma lentamente progressiva. Tem seu desenvolvimento na puberdade e progride durante as próximas duas décadas da vida, podendo afetar ambos os sexos e todos os grupos étnicos (WHEELER; HAUSER; AFSHARI; ALLINGHAM, 2013).

Embora seja difícil de acontecer, existem casos em que o acometimento unilateral se faz presente. Isso porque o ceratocone se caracteriza pelo acometimento bilateral, normalmente de forma assimétrica. Conforme a Figura 2, pode-se visualizar a diferença entre um olho normal X olho acometido pela ceratocone (LOPEZ; PINTO; SOUZA, 2015).

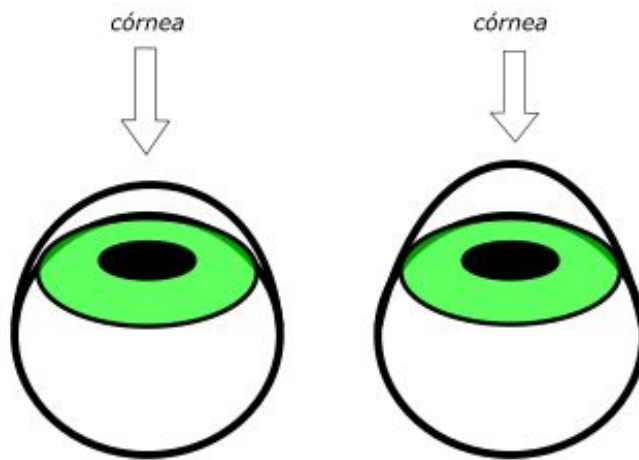


Figura 2 - Olho normal x Olho com ceratocone  
Fonte: (ROSA, 2019, p. 25)

### **2.1.3 Sintomas**

Os sintomas do ceratocone muitas vezes podem ser identificados em um exame de rotina. Porém, em alguns casos acaba sendo assintomático, conclusão que pode ficar ainda mais evidente quando se trata de uma fase inicial da doença (LAWLESS; COSTER; PHILLIPS; LOANE, 1989).

Os principais sintomas do ceratocone são apresentados, inicialmente, com a necessidade constante de troca de óculos ou uma menor tolerância ao uso de lentes de contato. Sendo possível, em fases mais avançadas, alcançar situações em que a visão se torna distorcida ou borrada, tanto para perto quanto para longe. Outros exemplos de sintomas são relatados, como: fotofobia, cansaço visual, irritação ocular e distorção nos reflexos de luz (LOPEZ; PINTO; SOUZA, 2015).

### **2.1.4 Diagnóstico**

Variadas são as formas possíveis de realização do diagnóstico do ceratocone. Entretanto, muitos sinais podem passar despercebidos no estágio inicial da doença. Segundo Lawless, Coster, Phillips e Loane (1989), com o objetivo de realizar o diagnóstico, é importante verificar a existência do afinamento da córnea ou astigmatismo irregular, uma vez que esses aspectos são classificados como condições mínimas para a existência do ceratocone. Topografia computadorizada da córnea, por sua vez, é um método de avaliação clínica e é utilizado para fases mais sensíveis da doença, quando o estágio ainda é inicial (WHEELER; HAUSER; AFSHARI; ALLINGHAM, 2013).

O mais preocupante, contudo, é que sintomas em formas subclínicas ou frustras podem passar despercebidos pelo médico e pelo paciente, além de que o quadro clínico, muitas vezes, pode ser confundido com doenças como miopia e astigmatismo irregular. Esses fatores, por sua vez, ressaltam a importância em realizar testes específicos para concluir o diagnóstico do

paciente (WHEELER; HAUSER; AFSHARI; ALLINGHAM, 2013; LOPEZ; PINTO; SOUZA, 2015).

### **2.1.5 Tratamento**

Ao realizar o tratamento do ceratocone, o principal objetivo é conseguir, por meio de uma forma simples, obter a normalidade visual. A maioria dos pacientes necessita de lentes de contato para solucionar o problema visual. Os óculos são suficientes para uma parcela menor de pacientes e, para a minoria, é fundamental a realização de uma cirurgia (LAWLESS; COSTER; PHILLIPS; LOANE, 1989).

Outra opção para realizar o tratamento do ceratocone é o procedimento criado na década de 50 e nomeado como implante de anel intracorneano. Esse procedimento é considerado menos invasivo e tem como objetivo realizar um aplanamento da curvatura corneana central.

Por outro lado, nem todos os casos de ceratocone são solucionados utilizando métodos comuns, como óculos, lentes de contato ou implantes de anel intracorneano. No último estágio da doença, é realizado o transplante de córnea, procedimento que consiste na substituição da córnea com a patologia por outra saudável (LOPEZ; PINTO; SOUZA, 2015).

## **2.2 Análise Paraconsistente**

Conhecida como lógica paraconsistente, essa engenharia de característica utiliza um sistema não clássico para lidar com situações conflitantes. Seu objetivo tem como foco principal transformar dados brutos, como, por exemplo, um vetor de características, em entradas relevantes, que, por sua vez, possibilita um melhor desempenho para determinado classificador (GUIDO, 2019).

O método, de maneira simplificada, é iniciado com o processo de normalização dos dados. Após esse passo, a continuação do processo é realizada por meio de duas fases: análise intraclasse e análise interclasses, que representam, respectivamente, as semelhanças e

dessemelhanças das características de uma determinada classe. Em outras palavras, pode-se afirmar que representam o nível de fé e descrença em determinadas características (GUIDO, 2019).

A análise intraclasse é quantificada por  $\alpha \geq 0$ , e a análise interclasse como  $\beta \leq 1$ . Portanto, sendo  $\alpha$  e  $\beta$  não complementares e representando o nível de fé e descrença, respectivamente, em um determinado conjunto de características, o uso de uma lógica comum não seria adequado para solucionar esse problema (GUIDO, 2019).

Ao iniciar a Análise Paraconsistente, realiza-se a análise intraclasse. Esta, por sua vez, utiliza a seguinte técnica para encontrar o parâmetro de desvio:  $Y = (1 - A)$ , sendo  $A$  correspondente à amplitude entre determinada característica de uma classe. Essa técnica corresponde a uma medida padronizada de semelhança, de maneira que  $Y \approx 0$  e  $Y \approx 1$  representam semelhança baixa e alta, respectivamente. Após esse método ser realizado em todas as classes, realiza-se a média de cada um dos  $Y$ , possibilitando, dessa forma, encontrar o alfa, que é identificado como o menor valor dentre todos (GUIDO, 2019).

Tal escolha é realizada visto que é a partir do menor valor que se consegue avaliar o pior caso. Esse procedimento pode também ser compreendido a partir das seguintes fórmulas:

$$Y(C_i) = \frac{1}{n} \sum_{k=1}^{n-1} svC_i[k]$$

onde:

$$\alpha = \min \{Y(C_1), Y(C_2) \dots Y(C_n)\}$$

A primeira fórmula representa a média aritmética entre os valores de um vetor de características. Já a segunda representa o valor mínimo das médias resultantes da equação anterior.

Em seguida, realiza-se a análise interclasse, que, no contexto, é denominada como uma medida de dessemelhança entre as características de determinada classe. Essa análise tem como ponto de partida a designação de uma classe principal e o armazenamento de dois vetores de

tamanho  $T$ , responsáveis por armazenar os maiores e menores valores de cada característica, respectivamente (GUIDO, 2019).

A partir dessa etapa, para definir o número total de sobreposições encontradas  $R$ , é realizada uma comparação entre os vetores que possuem os maiores e menores valores da classe principal com os vetores de características de todas as outras classes, com o intuito de computar a quantidade de vezes em que ocorre uma sobreposição. Esse procedimento é realizado com todas as classes, sendo possível, após essa fase, definirmos:

$$\beta = \frac{R}{F}$$

Onde  $F$  é o número de todas as sobreposições possíveis, ou seja,  $F = N * (N - 1) * X * T$  sendo  $N$  o número total de classes,  $X$  o número de vetores em cada classe e  $T$  o tamanho de cada vetor (GUIDO, 2019).

Após encontrar  $\alpha$  com base na análise intraclasse e  $\beta$  com base na análise interclasse, calculam-se os pontos  $G1$  e  $G2$ , responsáveis pelo grau de fé e descrença, respectivamente. Definindo  $G1 = \alpha - \beta$  e  $G2 = \alpha + \beta - 1$ , é possível calcular a distância do ponto  $P = (G1, G2)$  para o vértice  $(1,0)$  no plano paraconsistente, sendo assim, a menor distância classifica o melhor subconjunto das características originais para serem utilizadas no classificador, visto que o vértice  $(1,0)$  é considerado como o maior valor possível de fé em um conjunto de característica (GUIDO, 2019).

A Figura 3 ilustra o Plano Paraconsistente, onde o resultado da análise deve ser plotado:

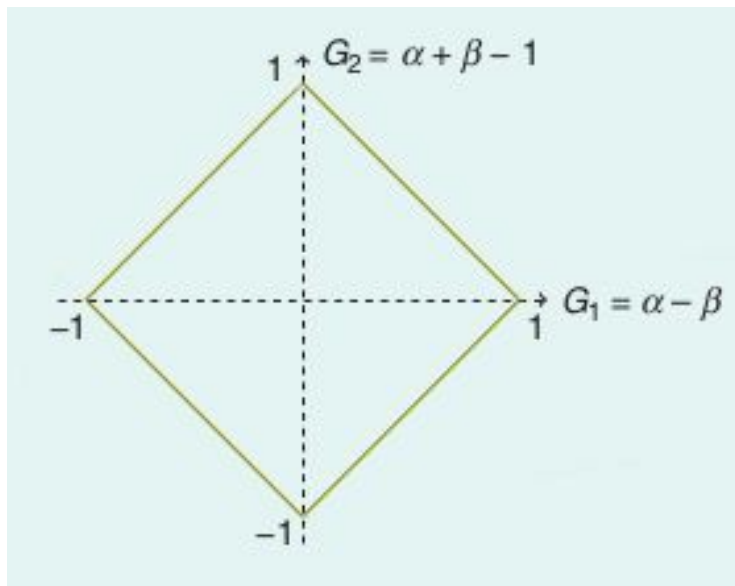


Figura 3 - Plano Paraconsistente.  
Fonte: (GUIDO, 2019, p. 4)

## 2.3 Inteligência Artificial

### 2.3.1 Introdução

Muito se tem falado sobre Inteligência Artificial, mas, de fato, o que é isso? Para Russel e Norvig (2013), existem 8 definições corretas para IA, separadas em duas dimensões:

- **Processos de pensamento e raciocínio:**

- O novo e interessante esforço para fazer os computadores pensarem, máquinas com mentes, no sentido total e literal. (HAUGELAND, 1985)
- Automatização de atividades que associamos ao pensamento humano, atividades como a tomada de decisões, a resolução de problemas e o aprendizado. (BELLMAN, 1978)
- O estudo das faculdades mentais pelo uso de modelos computacionais. (CHARNIAK; MCDERMOTT, 1985)
- O estudo das computações que tornam possível perceber, raciocinar e agir. (WINSTON, 1992)

- **Comportamento:**

- A arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas. (KURZWEIL, 1990)
- O estudo de como os computadores podem executar tarefas que, hoje, são melhor desempenhadas pelas pessoas. (RICH; KNIGHT, 1991)
- Inteligência Computacional é o estudo do projeto de agentes inteligentes. (POOLE; MACKWORTH; GOEBEL, 1998)
- Inteligência Artificial está relacionada a um desempenho inteligente de artefatos. (NILSSON, 1998)

Segundo Russel e Norvig (2013), as afirmações de Haugeland; Bellman; Kurzweil; e Rich e Knight, avaliam o funcionamento da IA em termos de fidelidade ao desempenho humano. Já as definições de Charniak e McDermott; Winston; Poole, Mackworth e Goebel; e Nilsson, medem a IA fazendo um comparativo com o conceito ideal de inteligência, definido por racionalidade. Todas as estratégias citadas anteriormente têm sido utilizadas durante a história por pessoas e em situações diferentes.

### **2.3.2 Área de atuação**

Inteligência Artificial é uma das áreas de estudos da Ciência da Computação, cujo objetivo é fazer as máquinas pensarem ou se comportarem de maneira inteligente - conforme as citações do tópico 2.3.1. É um campo de atuação muito amplo, que se ramifica em diversas outras áreas, correlacionadas à computação, ou não: psicologia, biologia, lógica matemática, linguística, engenharia, filosofia, entre outras (MONARD; BARANAUSKAS, 2000).

A ilustração a seguir demonstra algumas das áreas de estudo relacionadas à Inteligência Artificial:

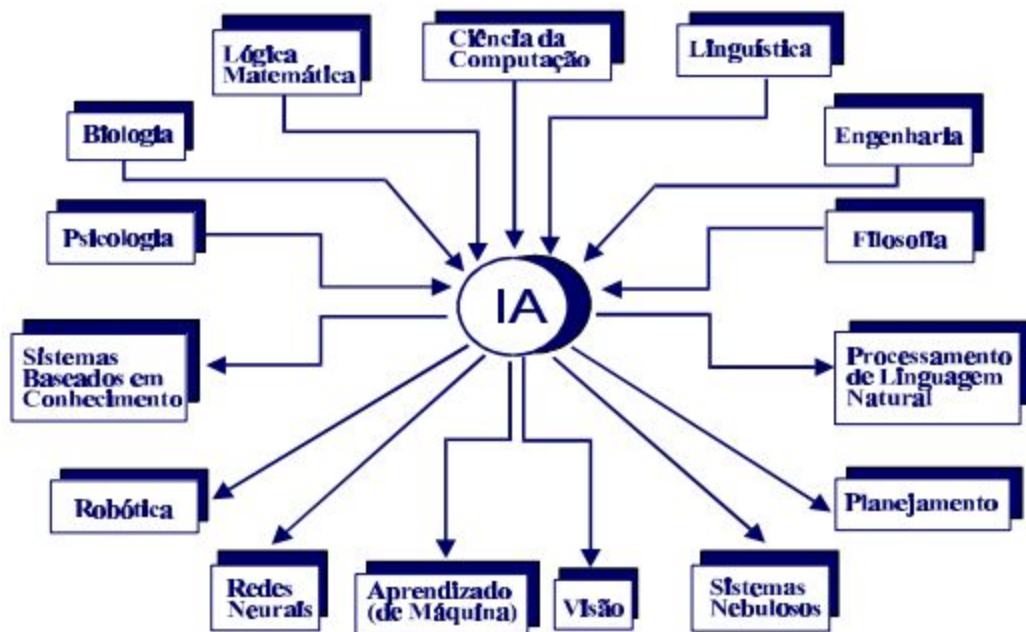


Figura 4 - Áreas relacionadas com IA  
 Fonte: (MONARD; BARANAUSKAS, 2000, p. 2)

## 2.4 Redes Neurais Artificiais

### 2.4.1 Introdução

O estudo de Redes Neurais Artificiais (RNAs) é um dos subcampos de pesquisa da Inteligência Artificial. São modelos de processamento computacional inspirados no modelo biológico humano, que possuem a capacidade de adquirir e manter o conhecimento por meio da exposição à informação. São representadas por conjuntos de neurônios artificiais, que são unidades de processamento computacionais, interligados por inúmeras conexões (análogas às sinapses neurais biológicas), produzidas por vetores e matrizes de “pesos sinápticos” (SILVA; SPATTI; FLAUZINO, 2019).

“As aplicações que envolvem a utilização de sistemas considerados inteligentes são as mais variadas possíveis” (SILVA; SPATTI; FLAUZINO, 2019, p. 21), por exemplo:

- Avaliação de imagens capturadas por satélites.



- Classificação de padrões de escrita e de fala.
- Reconhecimento de faces em visão computacional.
- Controle de trens de grande velocidade.
- Previsão de ações no mercado financeiro.
- Identificação de anomalias em imagens médicas.
- Identificação automática de perfis de crédito para clientes de instituições financeiras.

#### **2.4.2 Neurônio Biológico**

As RNAs seguem o modelo de funcionamento do cérebro humano. Esse, por sua vez, processa informações paralelamente, através dos neurônios, uma das principais células do sistema nervoso (SILVA; SPATTI; FLAUZINO, 2019).

Os neurônios têm a função de processar informações e, a partir disso, decidir se dispara ou não impulsos elétricos, as chamadas sinapses cerebrais. Sua estrutura é composta, principalmente, por três partes essenciais: dendritos, corpo celular e axônio (SILVA; SPATTI; FLAUZINO, 2019).

Os dendritos são finos e numerosos prolongamentos que têm a função de captar, continuamente, estímulos vindos de outros neurônios, ou do meio externo em que eles estiverem em contato - neurônios sensitivos (SILVA; SPATTI; FLAUZINO, 2019).

O corpo celular, também chamado de somador, é a parte responsável por processar todas as informações vindas dos dendritos e decidir se o neurônio poderá disparar um impulso elétrico ao longo do seu axônio (SILVA; SPATTI; FLAUZINO, 2019).

Por fim, o axônio é o responsável por conduzir os impulsos elétricos para outros neurônios conectores ou para aqueles que estão ligados diretamente com o tecido muscular. Sua

terminação também é formada por ramificações, chamadas de terminações sinápticas (SILVA; SPATTI; FLAUZINO, 2019).

As Figuras 5 e 6 ilustram, respectivamente, a estrutura de um neurônio biológico e as conexões sinápticas entre neurônios:

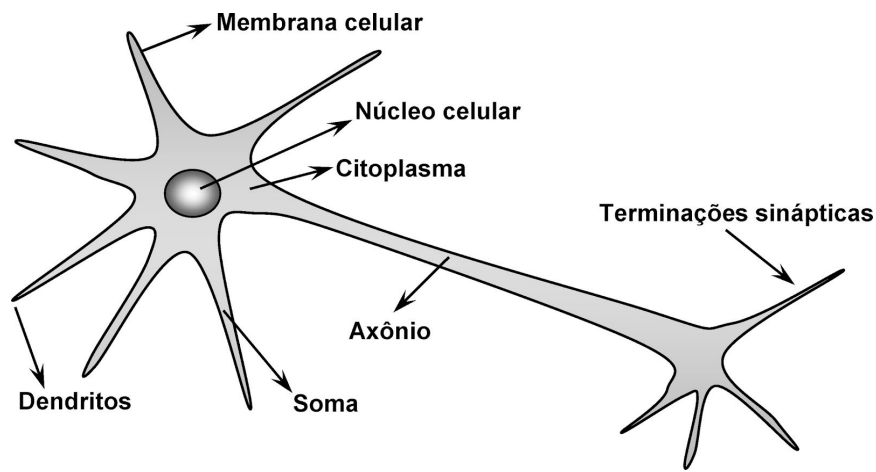


Figura 5 - Representação do neurônio biológico.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 29)

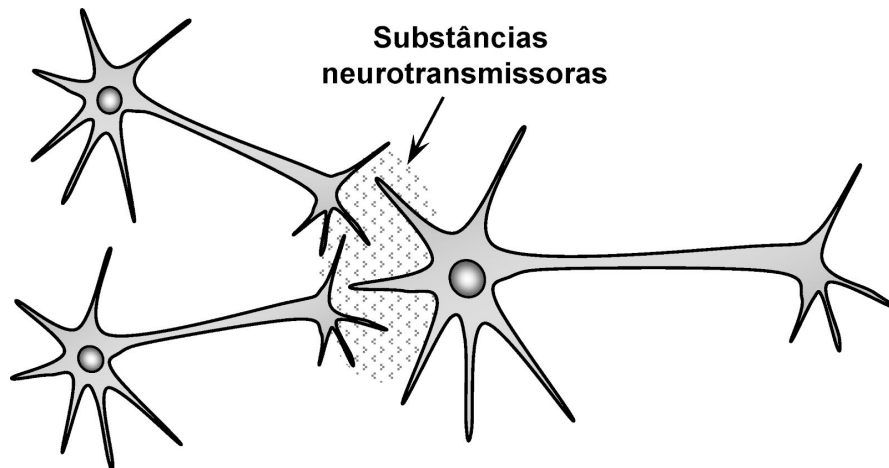


Figura 6 - Conexões sinápticas entre neurônios.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 30)

### 2.4.3 Neurônio Artificial

Segundo Silva, Spatti e Flauzino, a definição para neurônio artificial é a seguinte:

Os neurônios artificiais utilizados nos modelos de redes neurais artificiais são não-lineares, fornecendo saídas tipicamente contínuas, e realizam funções simples, como coletar sinais existentes em suas entradas, agregá-los de acordo com sua função operacional e produzir resposta, levando em consideração sua função de ativação inerente (SILVA; SPATTI; FLAUZINO, 2019, p. 33).

O funcionamento do neurônio artificial é análogo ao funcionamento do neurônio biológico, envolvendo suas características principais. Os impulsos elétricos captados pelos dendritos do neurônio biológico são equivalentes aos conjuntos numéricos  $\{x_1, x_2, x_3, \dots, x_n\}$  aplicados ao neurônio artificial. Já o corpo celular, que processa as informações vindas dos dendritos, é análogo às ponderações das entradas  $\{x_n\}$  em relação aos pesos sinápticos  $\{w_1, w_2, w_3, \dots, w_n\}$ , realizadas pelo neurônio artificial. A Figura 7 ilustra o funcionamento de um neurônio artificial:

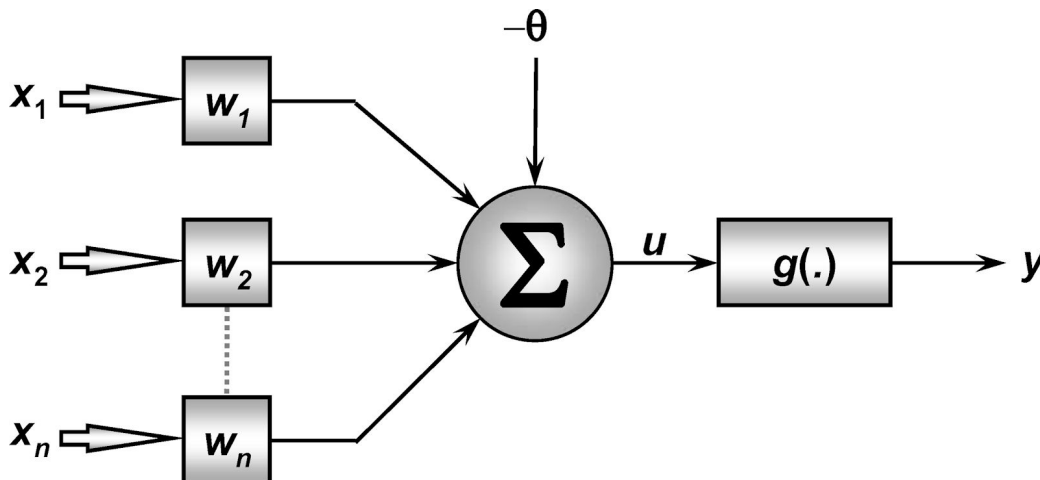


Figura 7 - Modelo do neurônio artificial.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 34)

Segundo Silva, Spatti e Flauzino (2019), podemos considerar que o neurônio artificial é formado por sete elementos essenciais:

- 1. Sinais de entrada**  $\{x_1, x_2, x_3, \dots, x_n\}$  : São valores do meio externo que representam variáveis de aplicação específica. Geralmente, são normalizados a fim de otimizar a eficiência dos algoritmos de aprendizagem.
- 2. Pesos sinápticos**  $\{w_1, w_2, w_3, \dots, w_n\}$  : Valores que qualificam os sinais de entrada de acordo com sua relevância. Geralmente, são iniciados de forma aleatória e ajustados durante o processo de treinamento do algoritmo.
- 3. Combinador linear**  $\{\Sigma\}$  : Realiza a soma de todos os valores ponderados (multiplicação de cada valor de entrada por seu respectivo peso), produzindo um valor de potencial de ativação.
- 4. Limiar de ativação**  $\{\theta\}$  : Valor que especifica o intervalo adequado para que o resultado do combinador linear possa gerar um valor de disparo na saída do neurônio artificial.
- 5. Potencial de ativação**  $\{u\}$  : Diferença entre o valor resultante do combinador linear e o limiar de ativação. Define se o neurônio produzirá um potencial de saída excitatório ou inibitório.
- 6. Função de ativação**  $\{g\}$  : Função que limita os valores de saída do neurônio dentro de patamares adequados, escolhida de acordo o tipo de problema analisado.
- 7. Sinal de saída**  $\{y\}$  : Resultado produzido pelo neurônio artificial, que poderá ser utilizado por outros neurônios que estiverem interligados.

O neurônio artificial proposto por Redes Neurais, princípios e práticas pode ser expresso pelas seguintes equações matemáticas:

$$u = \sum_{i=1}^n w_i \cdot x_i - \theta$$

$$y = g(u)$$

A primeira expressão representa o processo de aplicação dos sinais de entrada  $\{x_1, x_2, x_3, \dots, x_n\}$  e a multiplicação por seus respectivos pesos sinápticos  $\{w_1, w_2, w_3, \dots, w_n\}$  (processo de combinação linear  $\{\Sigma\}$ ), aplicação do limiar de ativação  $\{\theta\}$  e obtenção do potencial de ativação  $\{u\}$ .

Já a segunda expressão representa a aplicação de uma função de ativação específica  $\{g\}$  para obter o sinal de saída  $\{y\}$  do neurônio.

#### 2.4.4 Funções de Ativação

As funções de ativação limitam a saída de um neurônio, decidindo se ele é ativado ou não, ou seja, se sua saída possui relevância para o sistema. Elas podem ser de dois tipos principais: parcialmente diferenciáveis, que são aquelas que em as derivadas de primeira ordem são inexistentes em alguns pontos, e totalmente diferenciáveis, cujas derivadas de primeira ordem são conhecidas em todos os pontos (SILVA; SPATTI; FLAUZINO, 2019).

Ainda segundo os autores, dentre as funções parcialmente diferenciáveis, existem três tipos principais: função degrau, função degrau bipolar e função rampa simétrica, conforme descrito a seguir:

- 1. Função degrau:** resulta em valores unitários positivos quando o potencial de ativação do neurônio for nulo ou positivo; do contrário, resulta em valores nulos:

$$g(u) = \begin{cases} 1, & \text{se } u \geq 0 \\ 0, & \text{se } u < 0 \end{cases}$$

A Figura 8 demonstra a representação gráfica da função degrau:

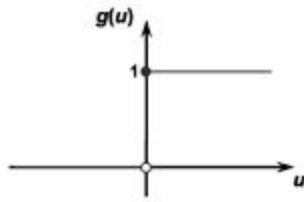


Figura 8 - Representação gráfica da função degrau.  
 Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 36)

- 2. Função degrau bipolar:** resulta em valores unitários positivos quando o potencial de ativação for maior que zero; valores nulos, se o potencial de ativação também for nulo; e valores unitários negativos, quando o potencial de ativação for menor que zero:

$$g(u) = \begin{cases} 1, & \text{se } u > 0 \\ 0, & \text{se } u = 0 \\ -1, & \text{se } u < 0 \end{cases}$$

A Figura 9 demonstra a representação gráfica da função degrau bipolar:

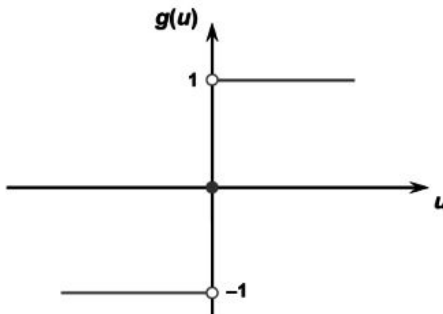


Figura 9 - Representação gráfica da função degrau bipolar.  
 Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 37)

- 3. Função rampa simétrica:** os resultados são iguais aos valores dos potenciais de

ativação quando estes estiverem no intervalo  $[-a, a]$ ; caso contrário, restringem-se aos valores limites:

$$g(u) = \begin{cases} a, & \text{se } u > a \\ u, & \text{se } -a \leq u \leq a \\ -a, & \text{se } u < -a \end{cases}$$

A Figura 10 demonstra a representação gráfica da função rampa simétrica:

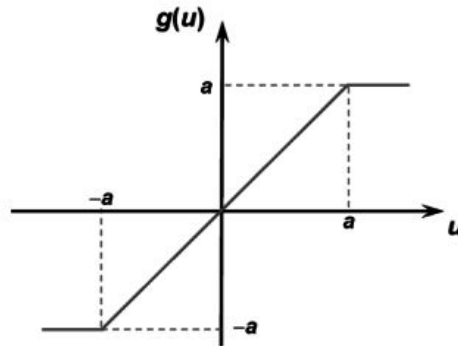


Figura 10 - Representação gráfica da função rampa simétrica.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 38)

No domínio das funções totalmente diferenciáveis existem quatro tipos principais: função logística, função tangente hiperbólica, função gaussiana e a função linear:

- 1. Função logística:** resulta sempre em valores reais entre zero e um, representados pela seguinte função matemática:

$$g(u) = \frac{1}{1 + e^{-\beta \cdot u}}$$

onde  $\beta$  é uma constante real associada ao nível de inclinação da função.

A Figura 11 demonstra a representação gráfica da função logística:

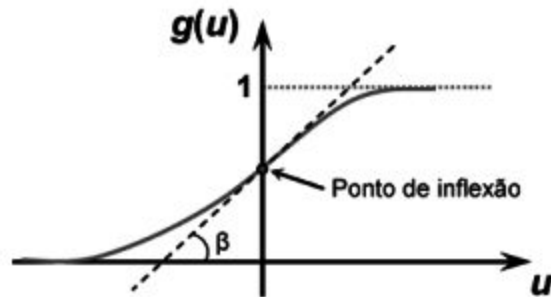


Figura 11 - Representação gráfica da função logística.  
 Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 39)

- 2. Função tangente hiperbólica:** seu funcionamento é semelhante ao da função logística, porém resulta em valores entre -1 e 1, conforme a seguinte equação:

$$g(u) = \frac{1 - e^{-\beta \cdot u}}{1 + e^{-\beta \cdot u}}$$

onde  $\beta$  é uma constante real associada ao nível de inclinação da função.

A Figura 12 demonstra a representação gráfica da função tangente hiperbólica:

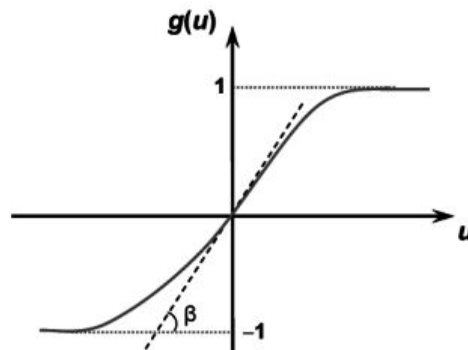


Figura 12 - Representação gráfica da função tangente hiperbólica.  
 Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 40)



- 3. Função gaussiana:** sua saída produz resultados iguais aos valores dos potenciais de ativação que estejam à mesma distância do seu centro, com a curva simétrica ao mesmo, conforme a equação:

$$g(u) = e^{-\frac{(u-c)^2}{2\sigma^2}}$$

onde  $c$  define o centro da função gaussiana e  $\sigma$  representa o desvio padrão.

A Figura 13 demonstra a representação gráfica da função gaussiana:

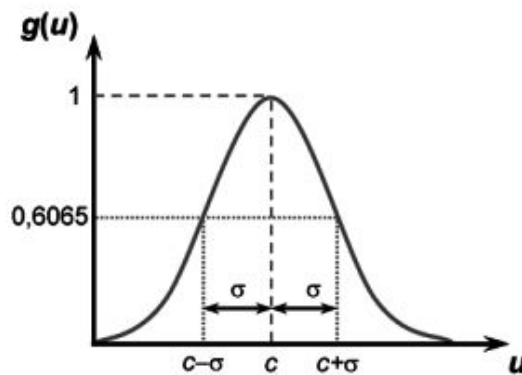


Figura 13 - Representação gráfica da função gaussiana.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 41)

- 4. Função linear:** produz resultados idênticos aos valores produzidos pelo potencial de ativação:

$$g(u) = u$$

A Figura 14 demonstra a representação gráfica da função linear:

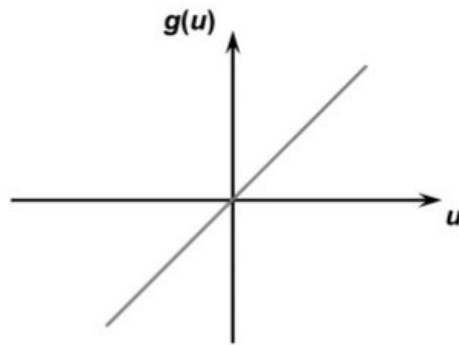


Figura 14 - Representação gráfica da função linear.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 42)

#### 2.4.5 Arquiteturas de Redes Neurais Artificiais

A arquitetura de uma rede neurais define a forma como os neurônios estão estruturados na rede, como eles estão interligados e como são organizadas as suas camadas, e está diretamente associada ao tipo de algoritmo de aprendizado utilizado para treinamento da mesma. (HAYKIN, 2007).

De acordo com Silva, Spatti e Flauzino (2019), uma rede neural pode ser dividida em três partes, chamadas de camadas. A primeira parte é a camada de entrada, que recebe os dados do meio externo e, geralmente, a fim de se obter o melhor desempenho do algoritmo, devem ser normalizados de acordo com a função de ativação utilizada. A segunda parte é composta por uma ou mais camadas de neurônios intermediários ocultos, que realizam o processamento dos dados advindos da primeira camada. Por fim, na terceira parte, existe a camada de saída, que é responsável por produzir o sinal de saída a partir do processamento das camadas anteriores.

As principais arquiteturas de redes podem ser classificadas como: redes *feedforward* de camadas simples, redes *feedforward* de camadas múltiplas e redes recorrentes (SILVA; SPATTI; FLAUZINO, 2019).

Na arquitetura *feedforward* de camada simples as informações seguem sempre um único fluxo, que é da camada de entrada em direção a camada de saída. As redes deste tipo geralmente são aplicadas na classificação de padrões e filtragem linear, utilizando em seus processos de

treinamentos algoritmos baseados na regra de Hebb ou na regra Delta. Dentre as redes mais comuns que empregam a arquitetura *feedforward* estão o *Perceptron* e o *Adaline*. (SILVA; SPATTI; FLAUZINO, 2019).

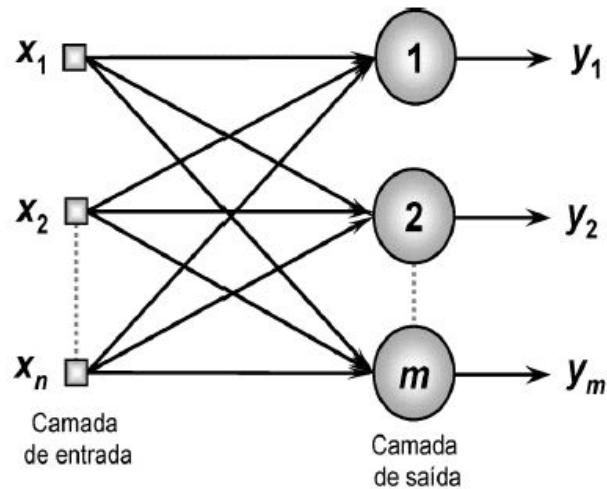


Figura 15 - Representação da arquitetura *feedforward* de camada simples.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 47)

A arquitetura *feedforward* de múltiplas camadas, diferentemente da arquitetura definida anteriormente, possui uma ou mais camadas de neurônios ocultos, a depender do tipo de problema a ser solucionado e a qualidade dos dados de entrada. Entre as principais redes que aplicam esta arquitetura, destacam-se o *Perceptron* multicamadas e as redes de base radial, que utilizam a regra Delta generalizada durante o processo de aprendizagem e podem ser aplicadas na aproximação de funções, classificação de padrões, dentre outras aplicações. (SILVA; SPATTI; FLAUZINO, 2019).

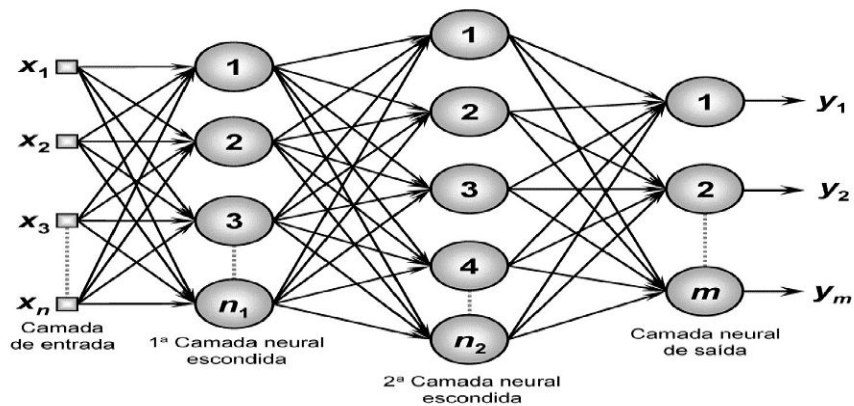


Figura 16 - Representação da arquitetura *feedforward* de múltiplas camadas.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 48)

A arquitetura recorrente utiliza a saída dos neurônios como sinais de entrada para outros neurônios, sendo qualificada para o processamento dinâmico de informações, aplicada em sistemas como a previsão de séries temporais. As redes de arquitetura recorrente utilizam algoritmos de aprendizagem baseados na minimização de funções de energia e na regra Delta generalizada, aplicadas principalmente nas redes Hopfield e *Perceptron* multicamadas (SILVA; SPATTI; FLAUZINO, 2019).

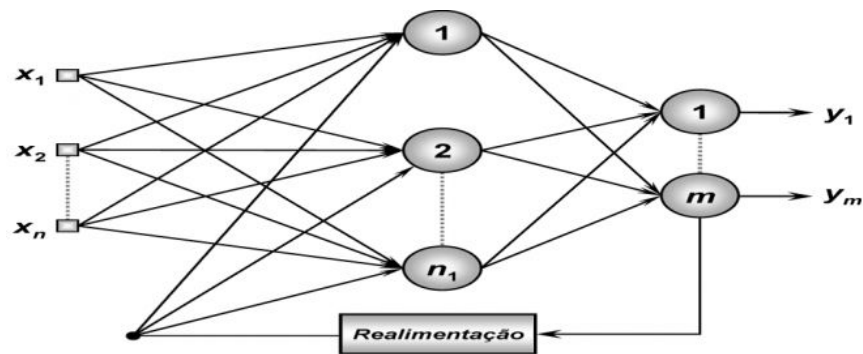


Figura 17 - Representação da arquitetura recorrente.  
Fonte: (SILVA; SPATTI; FLAUZINO, 2019, p. 49)

#### **2.4.6 Processos de Treinamento e Aprendizado**

Segundo Silva, Spatti e Flauzino (2019, p. 51), “O processo de treinamento de uma rede neural consiste da aplicação de passos ordenado que sejam necessários para sintonização dos pesos sinápticos e limiares de seus neurônios, tendo-se como objetivo final a generalização de soluções a serem produzidas pelas suas saídas”.

Os autores também afirmam que, geralmente, o conjunto de amostras é dividido em dois: subconjunto de treinamento, formado com cerca de 60 a 90% do conjunto original e utilizado exclusivamente durante o aprendizado da rede; e subconjunto de teste, formado com cerca de 10 a 40% do conjunto original e utilizado para teste de aceitação dos limiares e validação da topologia escolhida para solução do problema. Durante o processo de treinamento, cada apresentação dos valores do seu subconjunto para ajuste dos pesos da rede é denominado época de treinamento.

Existem diversos tipos de processos de treinamento como o treinamento não-supervisionado, treinamento com reforço, treinamento por lote de padrões e treinamento utilizando padrão-por-padrão (SILVA; SPATTI; FLAUZINO, 2019), mas estes serão ignorados por não se aplicarem ao escopo deste projeto, sendo considerado apenas os processo de treinamento supervisionado.

Ainda segundo Silva, Spatti e Flauzino (2019), o processo de treinamento supervisionado consiste em ter disponível o resultado esperado para cada amostra de treinamento apresentado ao algoritmo, comportando-se como se houvesse um “professor” instruindo a rede. Os autores concluem:

Os pesos sinápticos e limiares são então continuamente ajustados mediante a aplicação de ações comparativas, executadas pelo próprio algoritmo de aprendizagem, que supervisionam a defasagem entre as respostas produzidas pela rede em relação àquelas desejadas, sendo esta diferença usada no procedimento de ajuste. A rede será considerada treinada quando tal defasagem estiver dentro de valores aceitáveis, levando-se em consideração os propósitos de generalização de soluções.

### 2.4.7 Métricas de Desempenho

As métricas de desempenho avaliam os resultados produzidos pelas redes neurais, a fim de avaliar o quão confiáveis eles são e se o modelo utilizado é adequado para o problema em questão. Para a elucidação das métricas de avaliação, deve-se compreender as seguintes abreviações:

- **VP: Verdadeiro Positivo** – Exames classificados corretamente como patológicos.
- **VN: Verdadeiro Negativo** – Exames classificados corretamente como não patológicos.
- **FP: Falso Positivo** – Exames classificados incorretamente como patológicos.
- **FN: Falso Negativo** – Exames classificados incorretamente como não patológicos.

Com estes indicadores é possível montar a matriz de confusão, que compara os resultados reais esperados com as predições do classificador. O modelo da matriz de confusão pode ser observado na Tabela 1.

Tabela 1 - Modelo da matriz de confusão

Resultados Reais	Predições	
	Patológico	Não Patológico
	Patológico	Não Patológico
	Quantidade de VP	Quantidade de FP
	Quantidade de FN	Quantidade de VN

Fonte: autoria própria.

Para avaliar a eficiência do classificador, uma solução bastante utilizada é a Curva ROC – *Receiver Operating Characteristic*, que segundo Prati, Batista e Monard (2008, p. 1): “Ela é particularmente útil em domínios nos quais existe uma grande desproporção entre as classes ou quando se deve levar em consideração diferentes custos/benefícios para os diferentes

erros/acertos de classificação. Análise ROC também tem sido utilizada para a construção e refinamento de modelos”.

A Curva ROC pode ser representada por valores numéricos, ou plotada num gráfico, o que facilita a sua compreensão. A sua forma gráfica é baseada na taxa de verdadeiros positivos, plotada no eixo das ordenadas, e dos falsos positivos, plotados no eixo das abscisas. Um ponto no gráfico ROC é obtido do cálculo da taxa dos verdadeiros e falsos positivos a partir da matriz de confusão (PRATI; BATISTA; MONARD, 2008).

O gráfico ROC pode ser associado ao índice da área abaixo da curva (AUC) – *Area Under Curve*. O seu valor estará sempre entre 0 e 1, sendo 0 o valor péssimo e 1 o resultado ótimo, ou seja, espera-se que o classificador gere valores mais próximos de 1 (PRATI; BATISTA; MONARD, 2008).

### **3 METODOLOGIA E DESENVOLVIMENTO**

Nesta seção, são apresentados os detalhes dos métodos, estratégias e técnicas que foram empregados no planejamento, desenvolvimento e execução desta pesquisa, além dos meios utilizados para realizar a coleta dos dados e informações acerca do conteúdo prático e teórico do projeto.

#### **3.1 Metodologia Empregada**

A proposta deste trabalho é a realização de uma pesquisa de cunho descritivo para o desenvolvimento de um algoritmo computacional capaz de classificar tomografias oculares a fim de detectar o ceratocone, envolvendo os seguintes assuntos: Patologia Ceratocone, Normalização de Vetores, Análise Paraconsistente e Redes Neurais Artificiais. Essas pesquisas foram

realizadas por meio de livros e artigos, que proporcionam o conhecimento necessário para o desenvolvimento do projeto.

Os exames oftalmológicos utilizados para análise foram concedidos pelo Prof. Dr. Gildásio Castello de Almeida Jr, da Faculdade de Medicina de São José do Rio Preto – SP (FAMERP). Esses dados correspondem a 14 características para cada um dos exames de 177 pacientes aos quais tivemos acesso. Essas características foram obtidas por meio de Tomografias de Coerência Óptica. Os exames foram realizados na clínica Visum Centro de Olhos, localizada em São José do Rio Preto – SP, no período de julho de 2019 a junho de 2020.

### **3.2 Tecnologias Envolvidas**

O desenvolvimento prático deste projeto foi realizado através de computadores pessoais, assim como os demais recursos. Foi utilizada a linguagem de programação *Python 3.6*, implementado os códigos por meio do ambiente de desenvolvimento integrado *PyCharm*<sup>3</sup>. Todo o código foi versionado através do *Git* e publicado num repositório público no *GitHub*<sup>4</sup>.

#### **3.2.1 Ambiente de Desenvolvimento**

O ambiente de desenvolvimento utilizado para o desenvolvimento deste trabalho foi configurado em duas máquinas, um *notebook*, que possui processador *Intel Core i5 7th* com 4 núcleos de *2.5GHz*, memória *RAM* de *8GB*, *HDD* com *1TB* de armazenamento e sistema operacional *Microsoft Windows 10*; e um computador *desktop* com processador *AMD Ryzen 2200G* com 4 núcleos de *3.5 GHz*, memória *RAM* de *8GB*, *HDD* com *320GB* de armazenamento, *SSD* com *120GB de armazenamento* e sistema operacional *Linux Mint* versão 19 (*Tara*).

#### **3.2.2 Linguagem de Programação *Python***

---

<sup>3</sup> Disponível em: <https://www.jetbrains.com/pt-br/pycharm/download> Acesso em: 08 jul. 2020.

<sup>4</sup> Disponível em: <https://github.com/orgs/tcc-ads-2020/> Acesso em: 08 jul. 2020.



A linguagem de programação escolhida para a implementação algorítmica do projeto foi a linguagem Python, em sua versão 3.6. A escolha por essa linguagem aconteceu devido aos recursos nativos que facilitam o processo de desenvolvimento, por ser multiplataforma e de fácil portabilidade para o ambiente *Web*. Além disso, o *Python* conta com uma gama de bibliotecas disponíveis, como o *Matplotlib*<sup>5</sup>, que possibilita a geração de gráficos a partir de dados numéricos, de forma simples e rápida; e o *NumPy*<sup>6</sup>, que fornece um conjunto de funções que auxiliam a manipulação de vetores e a realização de cálculos numéricos.

### 3.2.3 *Pycharm Community Edition*

*PyCharm* é um ambiente de desenvolvimento integrado (IDE) desenvolvido pela empresa tcheca *JetBrains*, voltado especificamente para a linguagem *Python*. A princípio é um *software* pago, porém conta com uma versão comunitária, voltada para comunidade acadêmica, ou qualquer outra sem fins comerciais (JETBRAINS).

Este *software* foi escolhido por facilitar o processo de desenvolvimento em linguagem *Python*, pois conta com a documentação embutida, sugestões de complementação de código, integração com sistemas de controle de versão, interpretador e debugador, dentre outros recursos, além de ser gratuito.

### 3.2.4 Sistema de Controle de Versões - *Git*

Optamos por utilizar um Sistema de Controle de Versões (SCV) para garantir a segurança do código desenvolvido e manter o histórico de evolução do projeto. Segundo Chacon e Straub (2020, p. 9, tradução nossa)<sup>7</sup>: “Controle de versão é um sistema que registra alterações em um arquivo ou conjunto de arquivos ao longo do tempo, para que você possa recuperar versões específicas mais tarde.”

---

<sup>5</sup> Disponível em: <https://matplotlib.org/>. Acesso em: 08 jul. 2020.

<sup>6</sup> Disponível em: <https://numpy.org/>. Acesso em: 08 jul. 2020.

<sup>7</sup> *Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.*

*Git* é um dos SCV mais populares da atualidade devido a sua agilidade e facilidade de manipulação de *branches* (ou “ramificações”), e por isso optamos por sua utilização.

A maioria das operações no *Git* precisa apenas de arquivos e recursos locais para operar - geralmente nenhuma informação é necessária em outro computador na sua rede. Se você está acostumado a um SCV em que a maioria das operações tem sobrecarga de latência de rede, esse aspecto do *Git* fará você pensar que os deuses da velocidade abençoaram o *Git* com poderes não-mundanos. Como você tem todo o histórico do projeto no seu disco local, a maioria das operações parece quase instantânea (CHACON; STRAUB, 2020, p. 14, tradução nossa)<sup>8</sup>.

### 3.2.5 *GitHub*

O *Git* atua no versionamento local do projeto, porém não garante segurança em caso de pane na estação de trabalho e impossibilita a contribuição de outras pessoas no projeto de forma remota. Por isso, existem diversos repositórios *online* que permitem hospedar projetos de forma gratuita, dentre eles o *GitHub*, pelo qual optamos utilizar.

Além de manter o projeto seguro, o *GitHub* permite manter repositórios públicos, para que assim outros estudantes também possam estudar, ou contribuir com o projeto futuramente.

O *GitHub* é o maior host único para repositórios *Git* e é o ponto central da colaboração para milhões de desenvolvedores e projetos. Uma grande porcentagem de todos os repositórios *Git* está hospedada no *GitHub*, e muitos projetos de código aberto o usam para hospedagem, rastreamento de problemas, revisão de código e outras coisas (CHACON; STRAUB, 2020, p. 165, tradução nossa)<sup>9</sup>.

---

<sup>8</sup> *Most operations in Git need only local files and resources to operate — generally no information is needed from another computer on your network. If you're used to a CVCS where most operations have that network latency overhead, this aspect of Git will make you think that the gods of speed have blessed Git with unworldly powers. Because you have the entire history of the project right there on your local disk, most operations seem almost instantaneous.*

<sup>9</sup> *GitHub is the single largest host for Git repositories, and is the central point of collaboration for millions of developers and projects. A large percentage of all Git repositories are hosted on GitHub, and many open-source projects use it for Git hosting, issue tracking, code review, and other things.*

### 3.2.6 Google Planilhas

*Google Planilhas*<sup>10</sup> é uma plataforma *online* e gratuita para manipulação de planilhas eletrônicas. Seu uso neste trabalho se deve à necessidade de padronizar as informações dos exames oftalmológicos obtidos para então transformá-los em arquivos de formato CSV<sup>11</sup>.

### 3.3 Desenvolvimento

O processo de desenvolvimento deste trabalho ocorre conforme ilustra o fluxograma a seguir e é detalhado nos tópicos seguintes:

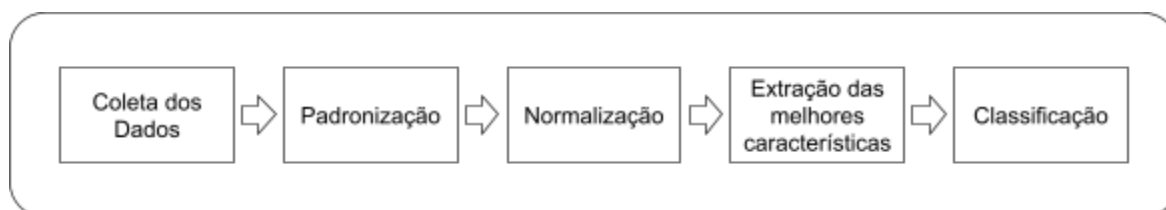


Figura 18 - Fluxo de desenvolvimento deste trabalho.  
Fonte: autoria própria.

#### 3.3.1 Coleta dos Dados

Os resultados das tomografias oculares (OCT) são gerados em formato de imagem, criando diversos arquivos com vários tipos de dados. Porém, apenas os dados sobre a espessura da córnea e do epitélio ocular são relevantes para este estudo. A Figura 19 ilustra o resultado de uma tomografia de um paciente anônimo:

<sup>10</sup> Disponível em: <https://www.google.com/intl/pt-BR/sheets/about/>. Acesso em: 08 jul. 2020.

<sup>11</sup> CSV é um formato de arquivo eletrônico e acrônimo para *Comma-separated values*, ou “valores separados por vírgulas” em português.

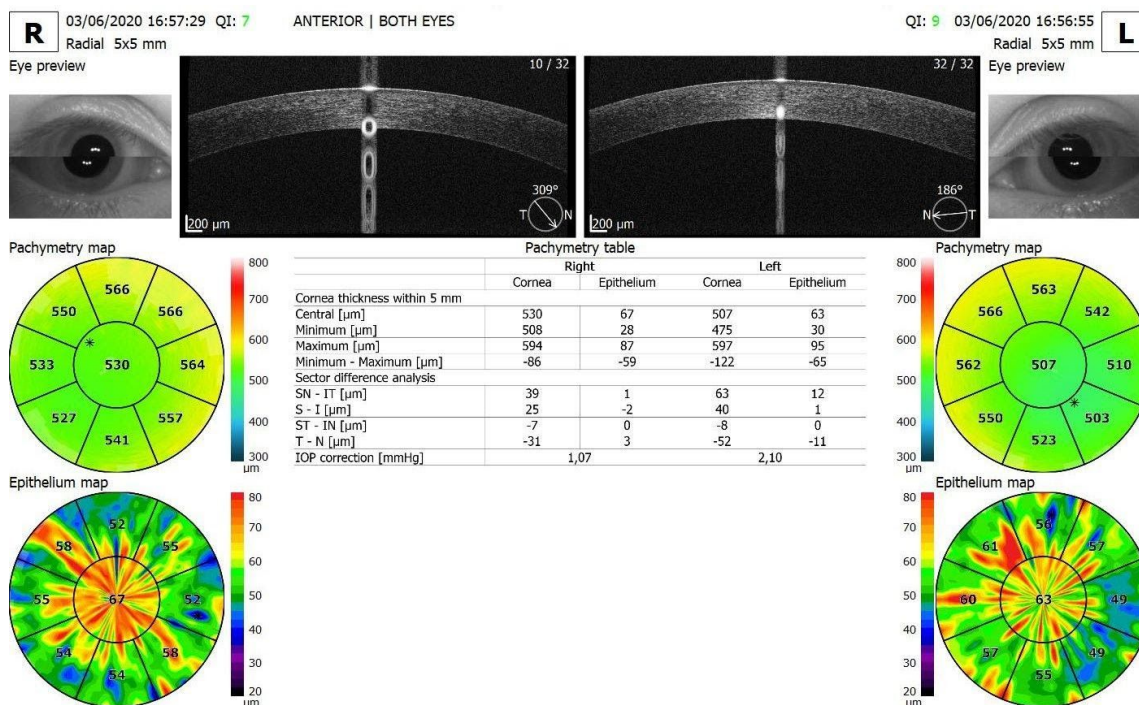


Figura 19 - Resultado de uma Tomografia de Coerência Óptica.  
Fonte: Fornecida pela clínica Visum Centro de Olhos.

### 3.3.2 Padronização

Da Tomografia de Coerência Óptica, os dados relevantes para o desenvolvimento deste projeto estão na tabela *Pachymetry table*, conforme é ilustrado na Figura 8 e pode ser visualizado mais facilmente na Figura 20.

Pachymetry table				
	Right		Left	
	Cornea	Epithelium	Cornea	Epithelium
Cornea thickness within 5 mm				
Central [ $\mu\text{m}$ ]	530	67	507	63
Minimum [ $\mu\text{m}$ ]	508	28	475	30
Maximum [ $\mu\text{m}$ ]	594	87	597	95
Minimum - Maximum [ $\mu\text{m}$ ]	-86	-59	-122	-65
Sector difference analysis				
SN - IT [ $\mu\text{m}$ ]	39	1	63	12
S - I [ $\mu\text{m}$ ]	25	-2	40	1
ST - IN [ $\mu\text{m}$ ]	-7	0	-8	0
T - N [ $\mu\text{m}$ ]	-31	3	-52	-11
IOP correction [mmHg]	1,07		2,10	

Figura 20 - *Pachymetry table*.  
Fonte: autoria própria.

A tabela ilustrada na Figura 20 contém as medidas das espessuras da córnea e epitélio ocular dos olhos direito e esquerdo do paciente examinado. Os atributos relevantes para análise são os seguintes:

- *Central*: espessura central da córnea e/ou epitélio ocular (RASHDAN, SHAH, ROBERTSON, 2019).
- *Minimum*: espessura mínima da córnea e/ou epitélio ocular (RASHDAN, SHAH, ROBERTSON, 2019).
- *Maximum*: espessura máxima da córnea e/ou epitélio ocular (RASHDAN, SHAH, ROBERTSON, 2019).
- *SN - IT*: diferença entre a espessura nasal superior e a espessura temporal inferior (RASHDAN, SHAH, ROBERTSON, 2019).
- *S - I*: diferença entre as regiões superior e inferior (RASHDAN, SHAH, ROBERTSON, 2019).
- *ST - IN*: diferença entre a espessura temporal superior e a espessura nasal inferior (RASHDAN, SHAH, ROBERTSON, 2019).
- *T - N*: diferença de espessura epitelial temporal e nasal (RASHDAN, SHAH, ROBERTSON, 2019).

A partir dos exames coletados, foi necessária a “extração” manual dos dados das imagens e a criação de uma planilha eletrônica, onde cada linha corresponde ao resultado de uma OCT. Um fragmento da planilha pode ser observado na Figura 21.

FENOTIPO	CENTRAL CORNEAL THICKNESS	MIN	MAX	SN - IT	S - I	ST - IN	T - N	EPITHELIUM THICKNESS	MIN	MAX	SN - IT	S - I	ST - IN	T - N
1		550	538	589	12	5	-6	-14	55	53	57	1	1	0
0		561	549	619	15	7	-7	-17	61	60	65	1	1	1
0		491	477	526	8	-7	-16	-19	64	61	65	0	1	1
1		488	473	532	26	20	3	-17	61	59	62	-1	-1	0
0		488	473	532	21	12	-5	-17	56	53	58	0	0	0
1		487	459	531	32	27	8	-17	54	50	56	-1	0	-1
1		533	512	580	19	11	-4	-18	57	55	59	0	0	0
1		537	521	635	15	6	-6	-12	60	55	76	-2	1	0

Figura 21 - Fragmento da planilha de dados sobre ceratocone.

Fonte: autoria própria.

A coluna “Fenótipo” corresponde ao diagnóstico do paciente, onde o valor “1” significa a presença do ceratocone, e “0” a ausência. “Central Corneal Thickness” corresponde ao parâmetro “Central”, da tabela *Pachymetry*, referente a espessura da córnea, assim como as colunas seguintes. A partir de “Epithelium Thickness”, que corresponde ao parâmetro “Central” da tabela “*Pachymetry*”, os dados passam a ser referentes à espessura do epitélio.

Em seguida, os dados da tabela foram exportados para um arquivo em formato CSV, para que fosse possível a manipulação pelos algoritmos desenvolvidos. A Figura 22 ilustra os dados gerados em formato CSV:

```

FENOTIPO,CENTRAL CORNEAL THICKNESS,MIN,MAX,SN - IT,S - I,ST - IN,T - N,EPI-THELIUM THICKNESS,MIN,MAX,SN - IT,S - I,ST - IN,T - N
1,550,538,589,12,5,-6,-14,55,53,57,1,1,0,0
0,561,549,619,15,7,-7,-17,61,60,65,1,1,1,0
0,491,477,526,8,-7,-16,-19,64,61,65,0,1,1,0
1,488,473,532,26,20,3,-17,61,59,62,-1,-1,-1,0
0,488,473,532,21,12,-5,-17,56,53,58,0,0,0,0
1,487,459,531,32,27,8,-17,54,50,56,-1,0,-1,0
1,533,512,580,19,11,-4,-18,57,55,59,0,0,0,0
1,537,521,635,15,6,-6,-12,60,55,76,-2,1,0,-1

```

Figura 22 - Dados sobre ceratocone exportados em formato CSV.

Fonte: autoria própria.

### 3.3.3 Normalização

Com o arquivo CSV pronto para ser manipulado é necessário normalizar os dados gerados entre o intervalo de zero e um, porém mantendo a integridade dos seus valores. Assim, é possível a obtenção das melhores características por meio da Análise Paraconsistente.

Para realizar a normalização dos dados, é necessário encontrar o maior e menor valor de todo o conjunto numérico. Em seguida, todo o conjunto deve ser subtraído pelo menor valor e, na sequência, dividido pela diferença entre o maior e menor valor. O processo de normalização pode ser descrito através da equação a seguir:

$$\frac{x - \min(\text{conjunto})}{\max(\text{conjunto}) - \min(\text{conjunto})}$$

A Figura 23 demonstra a saída do algoritmo normalizador, aplicado sobre os valores ilustrados na Figura 22:

```

[[ 0.03058104 0.87003058 0.85168196 0.92966361 0.04740061 0.03669725
  0.01987768 0.00764526 0.11314985 0.11009174 0.11620795 0.03058104
  0.03058104 0.02905199 0.02905199]
 [ 0.02905199 0.88685015 0.86850153 0.97553517 0.05198777 0.03975535
  0.01834862 0.0030581 0.12232416 0.12079511 0.12844037 0.03058104
  0.03058104 0.03058104 0.02905199]
 [ 0.02905199 0.77981651 0.75840979 0.83333333 0.0412844 0.01834862
  0.00458716 0. 0.12691131 0.12232416 0.12844037 0.02905199
  0.03058104 0.03058104 0.02905199]
 [ 0.03058104 0.77522936 0.75229358 0.84250765 0.06880734 0.05963303
  0.03363914 0.0030581 0.12232416 0.11926606 0.12385321 0.02752294
  0.02752294 0.02752294 0.02905199]
 [ 0.02905199 0.77522936 0.75229358 0.84250765 0.06116208 0.04740061
  0.02140673 0.0030581 0.1146789 0.11009174 0.117737 0.02905199
  0.02905199 0.02905199 0.02905199]
 [ 0.03058104 0.77370031 0.73088685 0.84097859 0.07798165 0.07033639
  0.0412844 0.0030581 0.1116208 0.10550459 0.1146789 0.02752294
  0.02905199 0.02752294 0.02905199]
 [ 0.03058104 0.8440367 0.81192661 0.91590214 0.05810398 0.04587156
  0.02293578 0.00152905 0.11620795 0.11314985 0.11926606 0.02905199
  0.02905199 0.02905199 0.02905199]
 [ 0.03058104 0.85015291 0.82568807 1. 0.05198777 0.0382263
  0.01987768 0.01070336 0.12079511 0.11314985 0.14525994 0.02599388
  0.03058104 0.02905199 0.02752294]]

```

Figura 23 - Dados sobre ceratocone normalizados.

Fonte: autoria própria.

### 3.3.4 Extração das melhores características

Após a normalização dos dados, é necessário realizar a extração das melhores características através da Análise Paraconsistente. Nesse processo, os coeficientes  $\alpha$  e  $\beta$  são encontrados, conforme está descrito na seção 2.2, obtendo o grau de certeza,  $G1 = \alpha - \beta$ , e o grau de contradição,  $G2 = \alpha + \beta - 1$ , sobre o conjunto analisado.

Para definir a relevância do conjunto de dados, é necessário efetuar testes com diferentes combinações de características da córnea e do epitélio ocular e avaliar a distância do ponto  $P(G1, G2)$ . Se uma característica for retirada do conjunto e  $P(G1, G2)$  se aproximar do ponto  $(0, 1)$ , significa que essa característica pode não ter muita relevância para a classificação. Caso contrário, se  $P(G1, G2)$  se distanciar do ponto  $(0, 1)$ , essa característica é potencialmente relevante.

A seleção das melhores características, por meio da utilização da Lógica Paraconsistente, tem início partindo da execução do algoritmo desenvolvido, com todas as características, registrando-se o resultado. Em sequência, foi retirada uma característica por vez, registrando o resultado. Por fim, foram retiradas as características que continham o índice ímpar e depois retiradas as características que continham o índice par.

Com o registro das combinações realizadas, tornou-se possível verificar algumas características que influenciavam diretamente para um resultado melhor ou pior. A partir desta análise foi possível verificar o melhor conjunto de características, sendo retirada as características que diminuem a distância de  $P(G1, G2)$  ao ponto  $(0, 1)$  do plano paraconsistente.

Por meio da realização de verificações manuais, algumas características referentes a córnea – *Central*, *Minimum*, *SN - IT*, *S - I*, *ST - IN* e *T - N* – foram retiradas do conjunto, enquanto que todas características referentes ao epitélio permaneceram. Com esse conjunto foi obtida a menor distância para ponto  $(0, 1)$  do plano paraconsistente: 0,202, conforme a Figura 24, onde está ilustrado os resultados obtidos através da análise paraconsistente.

```
Alfa: 0.859
Beta: 0.024

P=(G1, G2) = (0.835, -0.117)

Distância de P para o ponto (-1, 0) = 1.839
Distância de P para o ponto (1, 0) = 0.202
Distância de P para o ponto (0, -1) = 1.215
Distância de P para o ponto (0, 1) = 1.395
```

Figura 24 - Resultado final da análise paraconsistente.  
Fonte: autoria própria.

A fim de comparação, a Figura 25 ilustra os resultados da análise paraconsistente com todas as características originais.



```
Alfa: 0.748
Beta: 0.977

P=(G1, G2) = (-0.229, 0.726)

Distância de P para o ponto (-1, 0) = 1.059
Distância de P para o ponto (1, 0) = 1.427
Distância de P para o ponto (0, -1) = 1.741
Distância de P para o ponto (0, 1) = 0.357
```

Figura 25 - Resultado final da análise paraconsistente.  
Fonte: autoria própria.

O conjunto original possui 14 características ao todo, com a aplicação da lógica paraconsistente 6 delas foram retiradas, restando 8 características que formam o conjunto final para aplicação no classificador, conforme elencado na Tabela 2.

Tabela 2 - Características que serão utilizadas no classificador

Córnea	Epitélio						
<i>SN - IT</i>	<i>Central</i>	<i>Minimum</i>	<i>Maximum</i>	<i>SN - IT</i>	<i>S - I</i>	<i>ST - IN</i>	<i>T - N</i>

Fonte: autoria própria.

### 3.3.5 Classificação

Para realizar a classificação dos vetores de características foi utilizada a biblioteca *scikit-learn*<sup>12</sup>, disponível para linguagem *Python*. Foram testados o algoritmo de rede neural *Perceptron* Multicamadas<sup>13</sup> (PMC), treinado com *backpropagation*, comparando-o com a

<sup>12</sup> <https://scikit-learn.org/stable/index.html>  
<sup>13</sup> [https://scikit-learn.org/stable/modules/neural\\_networks\\_supervised.html#classification](https://scikit-learn.org/stable/modules/neural_networks_supervised.html#classification)

*Support Vector Machine*<sup>14</sup> (SVM) – Máquina Vetor de Suporte, *K-Nearest Neighbors*<sup>15</sup> (KNN) – K-ésimos Vizinhos Próximos e *Random Forest*<sup>16</sup> – Floresta Aleatória.

Cada exame de OCT gera dois vetores de características, correspondentes aos olhos direito e esquerdo do paciente, resultando num conjunto com 354 vetores ao todo. Desse conjunto, cerca de 30% foi separado para a realização de testes dos classificadores por meio da biblioteca *scikit-learn*, através do método *train\_test\_split*<sup>17</sup>, que busca equilibrar a proporção das classes para treinamento e testes. O conjunto formado para treinamento dos classificadores contém 247 vetores de características, sendo 207 saudáveis e 40 patológicos, e o conjunto para testes 107 vetores, com 79 saudáveis e 28 patológicos

Devido ao desbalanceamento da quantidade de vetores não patológicos (majoritário) e patológicos (minoritário), os resultados produzidos pelos classificadores não devem ser bem avaliados por métricas que consideram apenas a quantidade de acertos do classificador, como a acurácia, mas sim por métricas que levem em consideração a quantidade de verdadeiros e falsos positivos e negativos, como a curva ROC, métrica de avaliação utilizada neste trabalho.

A rede neural foi estruturada com 3 neurônios em uma única camada oculta, com o máximo de 1000 iterações, utilizando a função de ativação logística. Os testes realizados resultaram em AUC = 0.87, conforme o gráfico da Figura 26. A matriz de confusão apresenta 79 VP, 2 FP, 11 FN e 17 VN, ilustrados na matriz de confusão da Figura 27.

---

<sup>14</sup> <https://scikit-learn.org/stable/modules/svm.html#classification>

<sup>15</sup> <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

<sup>16</sup> <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

<sup>17</sup> [https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.train\\_test\\_split.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html)

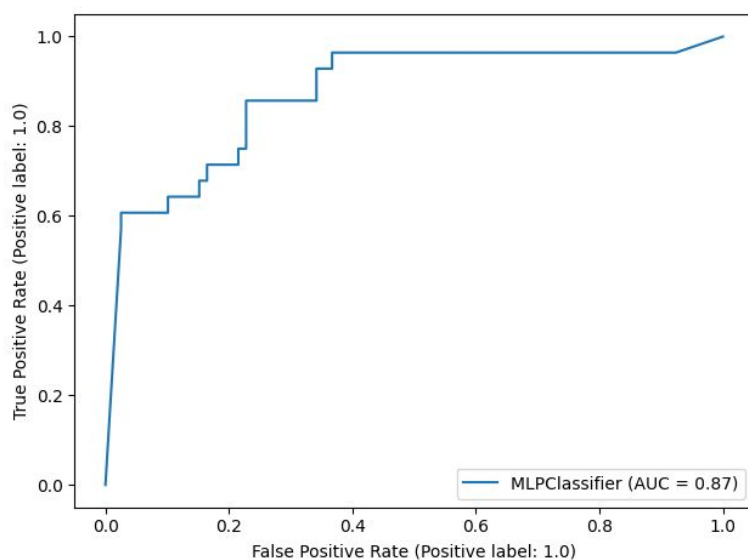


Figura 26 - Curva ROC do PMC.  
Fonte: autoria própria.

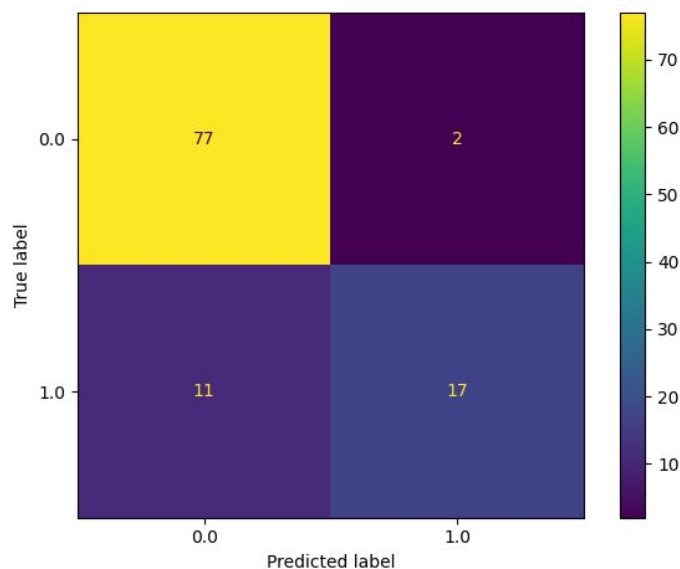


Figura 27 - Matriz de confusão do PMC.  
Fonte: autoria própria.

A Máquina Vetor de Suporte apresentou índices  $AUC = 0.85$ , conforme o gráfico da Figura 28. Sua matriz de confusão resulta em 79 VP, 0 FP, 16 FN e 12 VN, ilustrados na Figura 29. A comparação entre este e os próximos resultados serão apresentados na próxima sessão.

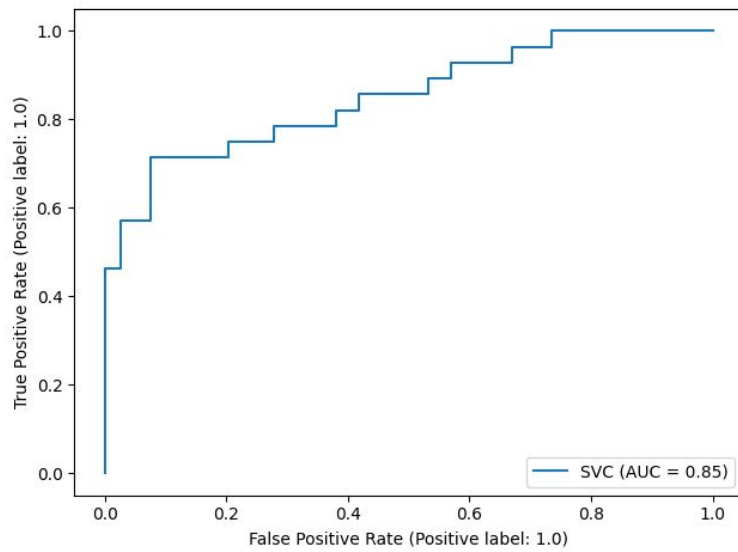


Figura 28 - Curva ROC da SVM.  
Fonte: autoria própria.

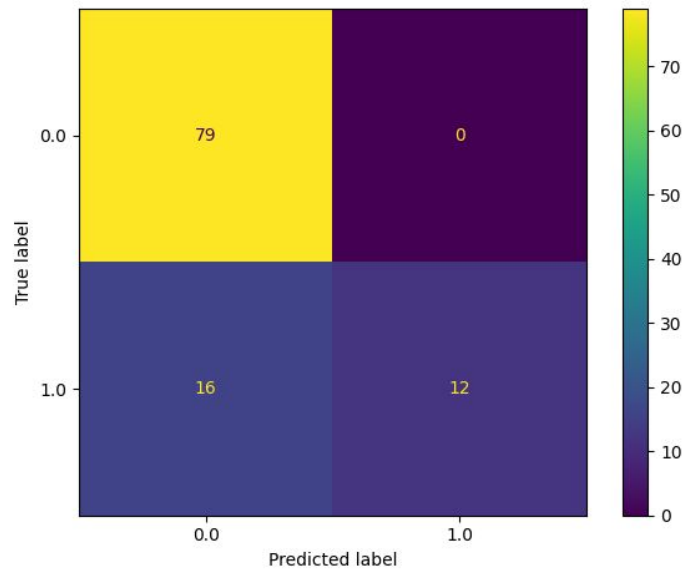


Figura 29 - Matriz de confusão da SVM.  
Fonte: autoria própria.

O algoritmo K-ésimos Vizinhos Próximos, comparando as características com os 7 vizinhos mais próximos, resultou no índice  $AUC = 0.8$ , como demonstrado no gráfico da Figura 30. A matriz de confusão apresenta 79 VP, 0 FP, 16 FN e 12 VN, ilustrados na Figura 31.

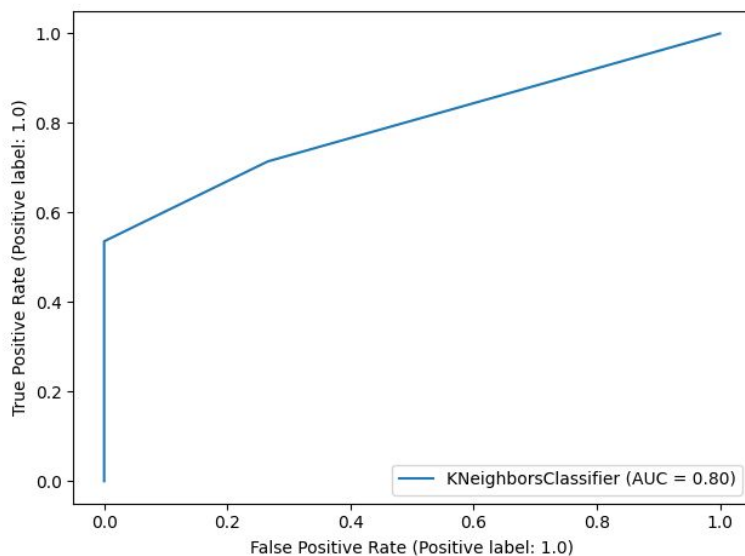


Figura 30 - Curva ROC do KNN.  
Fonte: autoria própria.

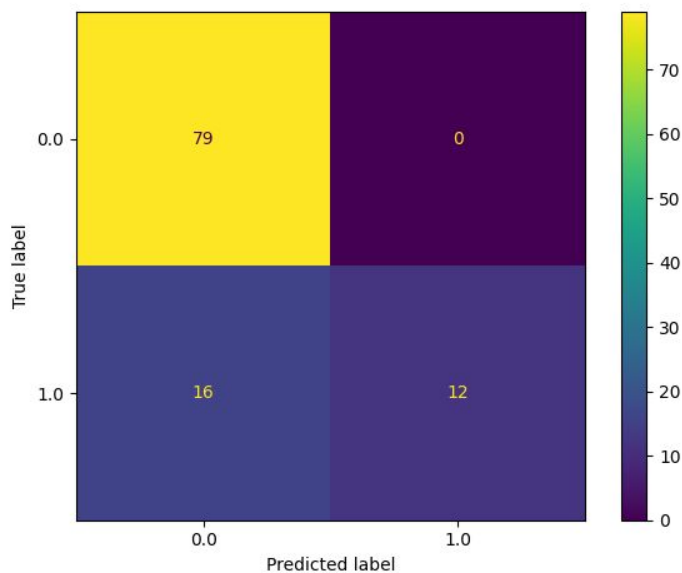


Figura 31 - Matriz de confusão do KNN.  
Fonte: autoria própria.

O último classificador testado, Floresta Aleatória, resultou em  $AUC = 0.82$ , demonstrado no gráfico da Figura 32. Sua matriz de confusão apresenta 79 VP, 0 FP, 16 FN e 12 VN, ilustrados na Figura 33.

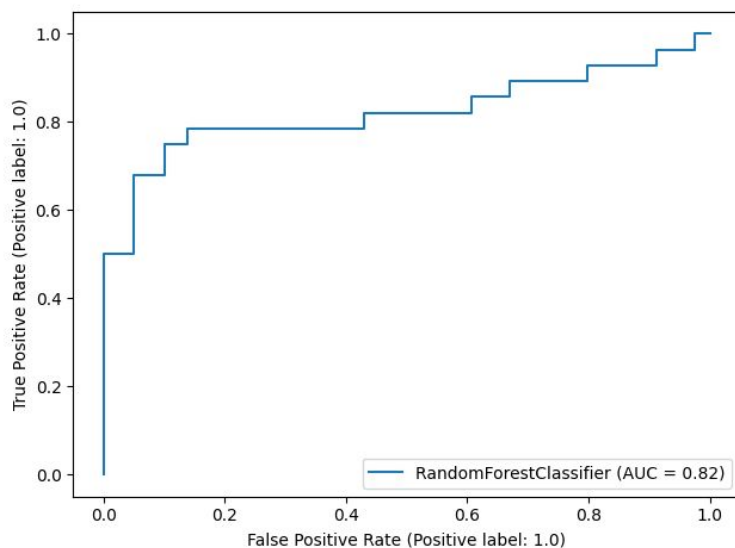


Figura 32 - Curva ROC da Floresta Aleatória.  
Fonte: autoria própria.

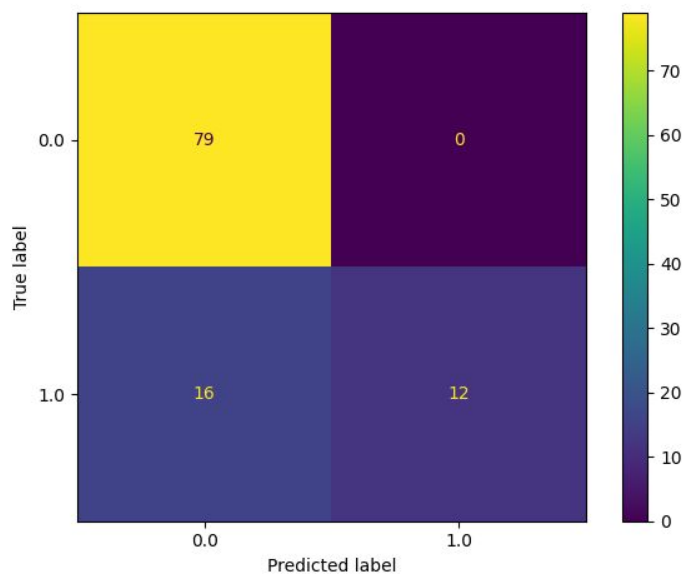


Figura 33 - Matriz de confusão da Floresta Aleatória.  
Fonte: autoria própria.

## 4 CONCLUSÃO

Tendo em vista o objetivo principal deste trabalho: desenvolver um algoritmo para classificar tomografias de coerência óptica através de dados da córnea e epitélio ocular, o resultado esperado foi atingido com sucesso. Comparando os resultados do *Perceptron* Multicamadas, solução mais complexa e robusta, com a Máquina Vetor de Suporte, *K-Nearest Neighbors* e Floresta Aleatória, soluções mais simples, a primeira solução apresenta maior eficiência na relação entre Verdadeiros Positivos e Falsos Positivos, demonstrada pela Curva ROC.

As matrizes de confusão dos classificadores mais simples apresentam resultados idênticos, enquanto que a rede neural apresenta 2 Falsos Positivos, contra 0 dos outros classificadores. Porém o *Perceptron* Multicamadas apresenta menos Falsos Negativos (11) e mais Verdadeiros Negativos (17), contra 16 e 12, respectivamente, dos outros algoritmos, demonstrando maior eficácia da rede neural.

As soluções mais simples apresentaram bons resultados, com destaque à SVM, porém a rede neural sobressai pela eficácia apresentada e demonstra que pode ter desempenho ainda melhor se for apresentada a conjunto de testes maiores. Além disso, a escolha dos algoritmos mais simples, embora possam ser aplicados para solução deste problema no lugar da rede neural, contando com os devidos ajustes, podem não ser eficientes num cenário com grande conjunto de características, visto que processam os dados em memória, ao invés de criar um modelo de aprendizado como na rede neural, e podem apresentar maiores custos de processamento.

### 4.1 Possíveis trabalhos futuros

Atividades futuras para extensão e aprimoramento deste trabalho podem ser realizadas, como:

- Estudo e tratamento específico do conjunto de características obtidas das OCTs, a fim de obter melhores resultados no classificador.

- Crescimento e otimização do processo de gerenciamento da base de dados, atualmente mantida em planilhas eletrônicas.
- Desenvolvimento de uma aplicação (web, móvel ou para desktops), que possibilite o uso do classificador por um profissional da área a fim de auxiliá-lo de fato na execução das suas tarefas.



## REFERÊNCIAS

BELLMAN, R. E. **An introduction to artificial intelligence**: can computers think? [s. l.]: Boyd & Fraser Publishing Company, 1978.

BUBACK, S. N. **Utilizando aprendizado de máquina para construção de uma ferramenta de apoio a moderação de comentários**. 2011. 65f. Dissertação - PUC-Rio - Pontifícia Universidade Católica do Rio de Janeiro. Rio de Janeiro. 2011.

CHACON, S.; STRAUB, B. **Pro Git**. 2. ed. [s. l.]: Apress, 2020. 521 p. Disponível em: <https://git-scm.com/book/en/v2>. Acesso em: 08 jul. 2020.

CHARNIAK, E.; MCDERMOTT, D. **Introduction to artificial intelligence**. [s. l.]: Addison-Wesley, 1985.

GRZYBOWSKI, A.; MCGHEE, C. N. J. The early history of keratoconus prior to Nottingham's landmark 1854 treatise on conical cornea: a review. **Clinical and Experimental Optometry**, Austrália, v. 96, n. 2, p. 140-5, 2013. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/23414219>. Acesso em: 26 maio 2020.

GUIDO, R. C. Paraconsistent feature engineering. **IEEE Signal Processing Magazine**, [s. l.], v. 36, n. 1, p. 154-158, jan. 2019. Disponível em: <http://hdl.handle.net/11449/186574>. Acesso em: 24 maio 2020.

HAUGELAND, J. **Artificial intelligence**: the very idea. Massachusetts: The MIT Press, 1985.

HAYKIN, S. **Redes neurais**: princípios e prática. 2. ed. Porto Alegre: Bookman, 2007. 898 p.

JETBRAINS. **PyCharm**: o IDE Python para desenvolvedores profissionais. Disponível em: <https://www.jetbrains.com/pt-br/pycharm>. Acesso em: 08 jul. 2020.

KURZWEIL, R. **The age of spiritual machines**. Massachusetts: The MIT Press, 1990.

LAWLESS, M.; COSTER, D. J., PHILLIPS, A. J., LOANE M. Keratoconus: diagnosis and management. **Australian and New Zealand Journal of Ophthalmology**, [s. l.], v. 17, n. 1, p. 33-60, fev. 1989. Disponível em: Acesso em: 26 maio 2020.

LOBO, L. C. Inteligência artificial, o Futuro da Medicina e a Educação Médica. **Revista Brasileira de Educação Médica**, Brasília, v. 42, n. 3, p. 3-8, set. 2018. Disponível em: <https://doi.org/10.1590/1981-52712015v42n3rb20180115editorial1>. Acesso em: 19 maio 2020.

LOPES do, A. C. do N.; PINTO, A. G. T.; DE SOUSA, B. A. Ceratocone: uma revisão. **Revista de Medicina e Saúde de Brasília**, Brasília, v. 4, n. 2, p. 219-232, maio-ago. 2015. Disponível em: <https://portalrevistas.ucb.br/index.php/rmsbr/article/view/5782/3969>. Acesso em: 26 maio 2020.

MONARD, M. C.; BARANAUSKAS, J. A. **Aplicações de inteligência artificial: uma visão geral**. Instituto de Ciências Matemáticas e de Computação de São Carlos. Disponível em: <http://dcm.ffclrp.usp.br/~augusto/publications/2000-laptec.pdf>. Acesso em: 31 maio 2020.

MOREIRA, L. B.; ALCHIERI, J. C.; BELFORT JÚNIOR, R.; MOREIRA, H. Aspectos psicossociais do paciente com ceratocone. **Arquivos Brasileiros de Oftalmologia**, São Paulo, v. 70, n. 2, p. 317-322, mar. 2007. Disponível em: <https://doi.org/10.1590/S0004-27492007000200023>. Acesso em: 25 maio 2020.

NILSSON, N. J. **Artificial intelligence: a new synthesis**. [s. l.]: Morgan Kaufmann, 1998.

PRATI, R. C.; BATISTA, G. E. A. P. A.; MONARD, M. C. Curvas ROC para avaliação de classificadores. **IEEE Latin America Transactions**, [s. l.], v. 6, n. 2, p. 215-222, 2008. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.do?arnumber=4609920&isnumber=4609907>. Acesso em: 03 jan. 2020.

POOLE, D.; MACKWORTH, A. K.; GOEBEL, R. **Computational intelligence: a logical approach**. [s. l.]: Oxford University Press, 1998.

RAMOS, A. **Fisiologia da visão: um Estudo Sobre o “ver” e o “enxergar”**. Pontifícia Universidade Católica do Rio de Janeiro. Disponível em: [http://sites.unifoa.edu.br/portal/plano\\_aula/arquivos/04054/Fisiologia%20da%20visao%20-%20MODULO%20I.pdf](http://sites.unifoa.edu.br/portal/plano_aula/arquivos/04054/Fisiologia%20da%20visao%20-%20MODULO%20I.pdf). Acesso em: 26 maio 2020.

RASHDAN, H.; SHAH, M.; ROBERTSON, D. M. The frequency of non-pathologically thin corneas in young healthy adults. **Clinical ophthalmology**, Auckland, v. 12, p. 1123-1135, jul. 2019. Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6628863> Acesso em: 11 set. 2020.

RICH, E.; KNIGHT, K. **Artificial intelligence**. 2. ed. [s. l.]: McGraw-Hill, 1991.

ROSA, J. M. **Análise paraconsistente de característica aplicada ao pré-diagnóstico de patologia subclínica**. 2019. 64 f. Monografia (Graduação em Ciência da Computação) - UNESP - Universidade Estadual Paulista. 2019.

RUSSEL, S. J.; NORVIG, P. **Inteligência artificial**. 3. ed. rev. ampl. São Paulo: Elsevier, 2013. 1016 p.

SILVA, I. N. da; SPATTI, D. H.; FLAUZINO, R. A. **Redes neurais artificiais para engenharia e ciências aplicadas: fundamentos teóricos e aspectos práticos**. 2. ed. rev. ampl. São Paulo: Artliber, 2019. 431 p.

WHEELER, J.; HAUSER, M. A.; AFSHARI, N. A; ALLINGHAM, R. R.; LIU, Y. **The genetics of keratoconus: a review.** **National Institutes of Health.** Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3686480/pdf/nihms-451306.pdf>. Acesso em: 26 maio 2020.

WINSTON, P. H. **Artificial intelligence.** 3. ed. [s. l.]: Addison-Wesley, 1992.