



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

ROMILDO DA SILVA FREITAS

**Desenvolvimento de modelo de rede neural convolucional
para classificação de cáries dentárias: abordagem usando
algoritmo genético como otimizador**

PATOS – PB

2018

ROMILDO DA SILVA FREITAS

**Desenvolvimento de modelo de rede neural convolucional
para classificação de cáries dentárias: abordagem usando
algoritmo genético como otimizador**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de bacharelado em Computação.

Orientadora: Dra. Kézia de Vasconcelos
Oliveira Dantas

PATOS - PB

2018

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

F866d Freitas, Romildo da Silva.

Desenvolvimento de modelo de rede neural convolucional para classificação de cáries dentárias [manuscrito] : abordagem usando algoritmo genético como otimizador / Romildo da Silva Freitas. - 2018.

51 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas , 2018.

"Orientação : Profa. Dra. Kézia de Vasconcelos Oliveira Dantas , Coordenação do Curso de Computação - CCEA."

1. Classificação de cáries. 2. Redes neurais convolucionais. 3. Algoritmos genéticos. 4. Diagnóstico odontológico. I. Título

21. ed. CDD 005.3

ROMILDO DA SILVA FREITAS

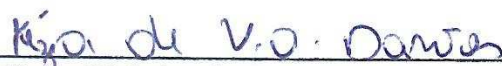
**Desenvolvimento de modelo de rede neural convolucional
para classificação de cáries dentárias: abordagem usando
algoritmo genético como otimizador**

Trabalho de Conclusão de Curso
apresentado ao Curso de Bacharelado em
Ciência da Computação da Universidade
Estadual da Paraíba, em cumprimento à
exigência para obtenção do grau de
bacharelado em Computação.

Área de concentração: Inteligência Artificial

Aprovada em: 28/12/2018.

BANCA EXAMINADORA



Dra. Kézia de Vasconcelos Oliveira Dantas
(Orientador)



Dr. Rodrigo Alves Costa
(Examinador)



Prof. Dr. Wellington Candeia de Araújo
(Examinador)

Dedico esse trabalho à minha mãe por todo o apoio durante essa caminhada. Que sempre deu os melhores conselhos e guiou-me aos caminhos corretos.

AGRADECIMENTOS

Primeiramente à Deus, por sempre me acompanhar nessa jornada, por nunca me deixar desistir mesmo diante das dificuldades e pela coragem de perseguir meus objetivos.

Aos meus pais, irmãos e a toda minha família que, com muito carinho e apoio, não mediram esforços para que eu chegasse até esta etapa de minha vida. Mãe, seu cuidado e dedicação foi que deram, em alguns momentos, a esperança para seguir. Pai, sua presença significou segurança e certeza de que não estou sozinho nessa caminhada.

Aos amigos e colegas, pelo incentivo e pelo apoio constantes.

Aos meus professores pelo conhecimento que me repassaram e pelos bons conselhos.

RESUMO

A cárie dentária é o problema bucal mais comum, atingindo cerca de 2,4 bilhões pessoas. Tem um tratamento simples em seu estágio inicial, no entanto, o problema se agrava bastante caso o diagnóstico seja tardio. Para auxílio ao diagnóstico, as radiografias são ferramentas indispensáveis, facilitando a identificação das cáries em locais difíceis de visualizar na análise clínica. Contudo, mesmo em posse de radiografias, a tarefa de identificação de lesões ainda é difícil. Para tanto, esse trabalho propõe o desenvolvimento de um modelo de rede neural convolucional capaz de classificar cáries em radiografias, assim tornando possível usá-lo para auxílio a identificação de cáries, o que, conseqüentemente, facilita o diagnóstico correto e diminui o tempo gasto em análises manuais. As redes neurais convolucionais são bastante eficazes em problemas de classificação, apresentando resultados muito satisfatórios em muitos casos. Um problema, no entanto, em definir um modelo de rede neural é encontrar uma arquitetura adequada, para solucionar esse problema, esta pesquisa fez uso de algoritmos genéticos como otimizadores dos hiperparâmetros de uma arquitetura base pré-definida, o que resultou em uma arquitetura otimizada, ou seja, adequada ao problema. Para treino e validação foi utilizado um conjunto de 697 amostras com e sem cáries, apresentando um resultado médio de 78.1% de acertos. Os resultados foram satisfatórios, em comparação aos atuais modelos existentes.

Palavras chaves: Classificação de cáries, Redes Neurais Convolucionais, Algoritmos Genéticos, Diagnóstico Odontológico

ABSTRACT

Tooth decay is the most common oral problem, reaching about 2.4 billion people. It has a simple treatment in its initial stage, however, the problem is aggravated greatly if the diagnosis is delayed. To aid diagnosis, radiographs are indispensable tools, facilitating the identification of caries in places difficult to visualize in the clinical analysis. However, even in possession of radiographs, the task of identifying lesions is still difficult. Therefore, this work proposes the development of a convolutional neural network model capable of classifying caries in radiographs, thus making it possible to use it to aid the identification of caries, which, consequently, facilitates the correct diagnosis and decreases the time spent in analysis. Convolutional neural networks are quite effective in classification problems, presenting very satisfactory results in many cases. A problem, however, in defining a neural network model is to find a suitable architecture to solve this problem, this research made use of genetic algorithms as optimizers of the hyperparameters of a predefined base architecture, which resulted in an optimized architecture, that is, appropriate to the problem. For training and validation, a set of 697 samples with and without caries was used, presenting an average result of 78.1% of correct answers. The results were satisfactory compared to the existing models.

Keywords: *Tooth decay classification, Convolutional Neural Network, Genetic Algorithms, Dental Diagnosis*

LISTA DE ILUSTRAÇÕES

Figura 1 Componentes de um neurônio biológico	22
Figura 2 Estrutura de um neurônio artificial.....	22
Figura 3 Rede neural MLP	24
Figura 4 Gráfico da função sigmóide, para valores de $a=1$, $a=2$ e $a = 4$	25
Figura 5 Modelo de treinamento supervisionado à esquerda e não-supervisionado à direita	26
Figura 6 Esquema de rede neural convolucional	29
Figura 7 Imagem e sua representação digital	30
Figura 8 Processo de convolução	31
Figura 9 Diferença entre maxpooling e average pooling	32
Figura 10 Diagrama do processo evolutivo utilizando algoritmos genéticos	35
Figura 11 Diagrama do processo de escolha do melhor modelo	37
Figura 12 Pré-processamento do dataset	38
Figura 13 Representação dos dados no processo de validação cruzada	41
Figura 14 Gráfico da acurácia nos dados de validação.....	43
Figura 15 Gráfico da função de perda nos dados de validação	44
Figura 16 Valores da acurácia (esquerda) e da função de perda (direita), para os dados de treinamento.....	44
Figura 17 Gráfico das acurácias para os dados de treinamento e de validação	45
Figura 18 Diagrama do modelo de CNN proposto	48

LISTA DE ABREVIATURAS

AG	Algoritmo Genético
CNN	<i>Convolutional Neural Network</i>
CAD	<i>Computer-aided Diagnostics</i>
ISBI	International Symposium on Biomedical Imaging
OMS	Organização Mundial da Saúde
RNA	Rede Neural Artificial

LISTA DE TABELAS

Tabela 1 Conjuntos de amostras.....	39
Tabela 2 Acurácia nos dados de validação na validação cruzada	46

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Cenário técnico científico	13
1.2	Problemática	16
1.3	Justificativa	16
1.4	Objetivos	17
1.4.1	Objetivo geral	17
1.4.2	Objetivos específicos	17
1.5	Metodologia	18
1.6	Estrutura do trabalho	19
2	FUNDAMENTAÇÃO TEÓRICA.....	20
2.1	Redes neurais artificiais	20
2.1.1	O neurônio biológico	20
2.1.2	O Neurônio Artificial	22
2.1.3	Redes Feedforward.....	23
2.1.4	Funções de ativação	24
2.1.5	Backpropagation	27
2.2	Rede neural convolucional (CNN)	28
2.2.1	Convolução	29
2.2.2	Downsampling.....	31
2.2.3	Arquitetura de uma CNN	32
2.3	Algoritmos Genéticos (AG)	33
2.3.1	Função de <i>fitness</i>	34
2.3.2	Critério de parada.....	34
2.4	Tecnologias	36
3	DESENVOLVIMENTO	37
3.1	Preparação do <i>dataset</i>	38
3.2	Escolha dos hiperparâmetros	39
3.3	Validação cruzada	40
3.4	Aplicação dos algoritmo genético	41
4	RESULTADOS	43
4.1	Modelo desenvolvido	46
5	CONSIDERAÇÕES FINAIS	49
	REFERÊNCIAS.....	50

1 INTRODUÇÃO

1.1 Cenário técnico científico

Cárie dentária é a principal doença bucal não transmissível, segundo a OMS (Organização Mundial de Saúde). Estudo realizado entre os anos de 1990 e 2010, pelo pesquisador Wagner Marcenes, aponta que cerca de 2,4 bilhões de pessoas em todo o mundo sofrem da doença. A cárie é causada pela produção de ácidos por bactérias residentes na boca. Ela causa a destruição do esmalte dentário, e o agravamento do problema pode chegar a destruir toda a coroa dentária.

É de suma importância que o problema seja identificado em seus estágios iniciais, desse modo permitindo que ela seja tratada e curada sem maiores danos ao dente, apenas um processo de remineralização é suficiente para a cura (BANDURA, 2018). Uma das formas de identificação de cáries, é através da análise de radiografias odontológicas. Segundo Rockenbach (2006), a radiografia é um exame complementar indispensável na detecção de cáries, uma vez que apenas a análise clínica pode deixar muitas lesões sem serem identificadas, principalmente as de tamanho reduzido. Bandura (2018) enfatiza que a análise visual e a radiografia são os métodos mais comuns de detecção de cáries, e que a segunda é especialmente boa para identificação de cáries interproximais, ou seja, aquelas que aparecem entre o contato de duas superfícies de dois dentes adjacentes, pois é difícil detectar somente a olho nu no exame clínico.

As cáries apresentam-se na radiografia com uma coloração escurecida, o que permite distinguir da coloração normal da coroa dentária. Contudo, nos estágios iniciais da doença essa identificação pode ser mais difícil, pois pode confundir-se com a coloração natural da coroa. Em outros casos a qualidade da radiografia pode dificultar a análise da mesma, principalmente radiografias mais antigas, como é o caso da radiografia convencional.

A radiografia convencional vem sendo utilizada desde 1895, quando Wilhelm Conrad Roentgen, fez a descoberta dos aparelhos de raios-X. Para a odontologia isso permitiu uma série de avanços no diagnóstico. Para a obtenção de radiografias

de boa qualidade, uma conjunto de fatores devem ser empregados, os quais podemos citar: aparelhos de raios-X, filtros, colimadores, filmes, processamento radiográfico e a técnica de tomada radiográfica. Durante muitos anos esse processo permaneceu quase inalterado, sofrendo apenas pequenas mudanças como filmes mais sensíveis, maior segurança tanto para operador quanto para paciente, entre outras mudanças (PEREIRA, 1999).

Nos últimos anos houve uma modernização no processo radiográfico, agora imagens digitais podem ser usadas ao invés de películas radiográficas convencionais, facilitando o armazenamento, processamento, análise, documentação e uma série de outras vantagens (PEREIRA, 1999). Além destas, a principal vantagem, é a grande redução da exposição do paciente à radiação, podendo chegar a 90%, uma vez que cargas menores de radiação podem ser aplicadas e através do processamento digital de imagens, conseguem-se radiografias com qualidade equivalente às convencionais. Outra grande vantagem é que as imagens podem passar por processos de realce de contraste e filtragem, o que possibilita aumentar significamente a detecção de cáries e a visualização da profundidade das mesmas (WATANABE, 1999).

Para White e Yoon (1997), mesmo com o uso das radiografias, o diagnóstico de cáries é uma tarefa muito difícil. Sendo que mesmo experientes dentistas costumam não identificar todas as lesões cariosas em um paciente. Møystad et al. (2003) citam a necessidade do desenvolvimento de métodos que possam diferenciar corretamente superfícies saudáveis de superfícies com lesões de cáries.

Com a introdução de imagens radiográficas digitais veio a possibilidade de introdução de sistemas CAD (*computer-aided diagnostics*), ferramentas para auxílio ao diagnóstico médico. Como citado em Watanabe (1999), através das imagens digitais, programas de computador podem ser utilizados para comparar duas imagens, o que possibilita determinar a progressão de doenças, assim como a avaliação do resultado do tratamento periodontal e endodôntico. Contudo, a gama de possibilidades vai muito além destas citadas até agora. Bandura (2018), por exemplo, cita que sistemas CAD podem ser utilizados não somente para detecção de cáries, mas também classificação do tipo, dependendo da profundidade da lesão. Além disso outro uso vem a ser a segmentação, que além de determinar a presença das cáries, também delimita a área da mesma.

Durante a última década, alguns sistemas baseados em aprendizado de máquina para classificação têm sido introduzidos no setor odontológico. Mesmo que algum progresso tenha sido obtido, a detecção de cáries em radiografias ainda é insuficiente. Em 2015, o *International Symposium on Biomedical Imaging (ISBI)* definiu a segmentação automática na radiografia dentária como um dos grandes desafios em análise de imagens dentárias e, de acordo com os resultados, o problema continua sem solução (BANDURA, 2018).

Algumas pesquisas desenvolvidas para dar suporte ao desenvolvimento de sistemas CAD seja para detecção, classificação ou segmentação, são apresentadas a seguir.

Ronneberger et al. (2015) apresentam em seu trabalho, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, uma arquitetura de rede neural convolucional denominada *U-net*, a qual foi utilizada para segmentação de regiões de interesse nas radiografias. O trabalho apresenta a segmentação automática das seguintes regiões dentárias: cárie, esmalte, dentina, polpa, coroa, restauração e tratamento de canal da raiz. Os resultados apresentados, no entanto, são insatisfatórios, uma vez que a detecção das cáries, o objetivo principal do desafio, apresenta uma precisão muito baixa, cerca de 0,078, ou seja, apenas 7,8% de precisão (WANG, et al., 2016).

Valizadeh et al. (2015), apresentam em seu trabalho, *Designing of a Computer Software for Detection of Approximal Caries in Posterior Teeth*, uma ferramenta desenvolvida para a detecção automática de lesões cariosas nos dentes posteriores (molares e premolares) e determinação da profundidade e posição das mesmas, pela análise digital das radiografias. Os autores informam que, em 25 superfícies com cáries no esmalte, um total de 15 foram identificadas, representando 60% de precisão. Enquanto que detectou 93 de 95 cáries localizadas na região interproximal da dentina, representado um acerto de 97%. Como citam os autores, o software apresenta bons resultados na detecção de lesões na dentina, mas apresenta dificuldade na detecção de cáries no esmalte.

Partindo do ponto que sistemas CAD são de tão grande relevância à odontologia, como uma forma de se obter uma segunda opinião, e que estes ainda carecem de grandes melhorias, pois os atuais sistemas apresentam resultados insatisfatórios, esse trabalho visa apresentar uma arquitetura de rede neural convolucional utilizada no contexto de classificação de cáries dentárias. Essa

arquitetura foi desenvolvida utilizando algoritmos genéticos, como otimizadores dos hiperparâmetros de uma arquitetura base, pré-definida.

1.2 Problemática

A identificação tardia das lesões causadas por cáries acarreta grandes danos aos dentes do paciente e com reparação muito mais complicada. No entanto, a detecção precoce da doença não é uma tarefa fácil, principalmente quando esta ocorre em regiões em contato entre dois dentes adjacentes. O uso da radiografia faz-se necessária nestes casos, pois somente a análise a olho nu, muitas vezes, não é capaz de identificar o problema. Porém, mesmo com a utilização de radiografias, a identificação da doença é difícil, pois a radiolucidez da região cariada é muito semelhante à da coroa.

Diante desses problemas, técnicas avançadas de processamento de imagens e visão computacional podem ser utilizadas para auxiliar os odontólogos na identificação do problema em seus estágios iniciais. Para tanto, sistemas CAD podem ser desenvolvidos para auxiliar o profissional na correta avaliação de uma região do dente. Contudo, esses ainda devem sofrer grandes avanços para auxiliarem de forma mais expressiva no diagnóstico da doença.

Os atuais algoritmos e técnicas de processamento e análise de imagens, ainda estão distantes de apresentar bons resultados no que se refere à identificação das cáries de forma automática. Técnicas mais sofisticadas já estão sendo utilizadas há alguns poucos anos, contudo, necessitam de mais pesquisas na área, uma vez que os resultados ainda são insatisfatórios e com baixa precisão.

1.3 Justificativa

Nesse trabalho será desenvolvida uma arquitetura de rede neural artificial para o problema de classificação de cáries dentárias. A partir dessa rede neural pode ser construído um sistema CAD para auxílio aos odontólogos na identificação de lesões cariosas na arcada dentária de seus pacientes. O objetivo dessa

arquitetura, após treinada, é classificar se dada uma região do dente a mesma apresenta ou não cáries.

Contudo a construção de uma rede neural é um trabalho muito experimental, uma vez que, para cada problema específico, pode ser necessária uma rede também específica. E a escolha da arquitetura é realizada através, muitas vezes, utilizando o método da tentativa e erro, ou seja, a escolha dos hiperparâmetros é feita de forma arbitrária, então o modelo é testado e novamente são feitas mudanças e assim sucessivamente. Esse método é utilizado, pois não existem heurísticas que forneçam um caminho rápido para chegar a um modelo ótimo. Existem métricas para testar o desempenho de um modelo, mas isso só após o seu treinamento.

Diante do problema de encontrar uma arquitetura ótima para o problemática em questão, optou-se pelo uso de algoritmos genéticos (AG). Essa abordagem possibilita que o AG faça o trabalho de encontrar os melhores hiperparâmetros para o modelo, poupando o pesquisador do trabalho de criar e testar modelos. Uma outra abordagem é através da força bruta, que consiste em usar um algoritmo para testar todas as combinações possíveis dentro de um conjunto limitado de hiperparâmetros, contudo, esse método é muito demorado tornando sua utilização pouco aconselhável.

1.4 Objetivos

Nesta seção são apresentados o objetivo geral desse trabalho e os objetivos específicos, os quais devem ser concluídos para que se possa chegar ao objetivo geral.

1.4.1 Objetivo geral

O objetivo geral desta pesquisa é desenvolver uma solução capaz de classificar corretamente cáries, através da análise de imagens de radiografias odontológicas. Dada uma região da radiografia, esta solução deve ser capaz de prever corretamente se a região apresenta ou não cáries.

1.4.2 Objetivos específicos

- Definir arquitetura base de rede neural convolucional;

- Selecionar conjunto de hiperparâmetros, que serão avaliados e escolhidos os mais adequados à rede neural;
- Obter base de imagens para treinando e teste da arquitetura;
- Realizar Pré-processamento das imagens obtidas;
- Aplicar algoritmos genéticos na otimização dos hiperparâmetros da rede neural;
- Analisar e expor os resultados obtidos.

1.5 Metodologia

Essa é uma pesquisa bibliográfica exploratória qualitativa que visa o desenvolvimento de uma arquitetura de rede neural convolucional (CNN - *convolutional neural network*) capaz de predizer se qualquer dada região de uma imagem de radiografia periodontal possui cáries. Nessa seção será apresentada a metodologia empregada para realizar o desenvolvimento desta arquitetura.

De antemão foi realizada uma pesquisa na bibliografia acerca do problema das cáries, da utilização de algoritmos de redes neurais para solução de problemas de classificação, assim como a utilização de algoritmos genéticos para solução de problemas de otimização em redes neurais.

O próximo passo foi o de reunir uma base de dados (*dataset*) de imagens radiográficas para utilização no processo de treinamento e validação da solução desenvolvida. Um *dataset* foi obtido de doutorandos do curso de odontologia da Universidade Estadual da Paraíba, os quais realizaram o laudo das imagens, demarcando as principais regiões de interesse presentes nas imagens, como: cáries, restaurações, obturações dentre outras. Esse *dataset* contém um total de 102 imagens.

O *dataset* passou por um pré-processamento, sendo realizada a extração de regiões das imagens com e sem a presença de cáries. De cada imagem foi realizada a extração das regiões que continham cáries, dando origem a uma nova imagem de amostra para o treinamento da CNN, assim como também a extração de regiões onde não apresentava lesões da doença, mas que eram facilmente confundíveis com a doença, uma vez que apresentava padrões e texturas muito semelhantes.

O próximo passo foi o de definir um modelo básico de CNN. Esse modelo apresenta lacunas, as quais são preenchidas através dos AGs. Sendo assim o mesmo só apresenta duas camadas bem definidas, a de entrada e a de saída. O resto do modelo é obtido somente como indivíduo do AG, quando serão escolhidos os genes, ou seja, os hiperparâmetros, e então montado o modelo (cromossomo ou indivíduo).

Durante o processo de evolução do AG os indivíduos são gerados aleatoriamente e a medida que as gerações avançam o processo torna-se menos aleatório. Esses indivíduos, ou seja, modelos de CNN, são os candidatos a solução do problema. Eles são treinados e avaliados com as amostras. Os melhores posicionados, que apresentam boa acurácia, são mantidos e se reproduzem, os demais são descartados.

Uma vez que o processo evolucionário termina, os melhores indivíduos são então testados novamente utilizando um processo de validação cruzada, para testar a qualidade de generalização do mesmo. Feito isso, o melhor modelo é selecionado, ou seja, o que apresentar melhor acurácia.

1.6 Estrutura do trabalho

O *capítulo 2* apresenta todo o embasamento teórico necessário para compreender os aspectos essenciais de redes neurais artificiais e algoritmos genéticos, e como os dois podem ser utilizados em conjunto para obtenção de resultados melhores.

O *capítulo 3* apresenta o processo de desenvolvimento utilizado, para a geração do modelo proposto.

O *capítulo 4* apresenta os resultados obtidos.

O *capítulo 5* apresenta as conclusões e objetivos futuros relacionados ao trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os fundamentos necessários para compreender o processo de desenvolvimento da arquitetura proposta. Serão apresentados conceitos que englobam processamento digital de imagens, aprendizado de máquina, redes neurais artificiais e algoritmos genéticos.

2.1 Redes neurais artificiais

Em 1987 o Instituto Americano de Engenharia Eletro-eletrônica convocou a primeira conferência em redes neurais. Em San Diego, Califórnia, 200 autores apresentaram seus artigos. Esse evento foi descrito como “o alvorecer da nova era”. Os cientistas estavam apresentando um tipo de computação inspirado pelas células neurais do cérebro humano. Essa nova forma de computação é definida como uma cultura interdisciplinar, a qual é baseada na forma como o cérebro aprende, o oposto da computação tradicional, que é fundamentada na programação (ALEKSANDER e MORTON, 1990).

Ainda em Aleksander e Morton (1990), a computação neural é definida como o estudo de redes de nós (neurônios) adaptáveis, os quais através de um processo de aprendizado (ou treinamento), por um conjunto de exemplos, é capaz de armazenar conhecimento experimental e torná-lo disponível para uso. Isso implica que a rede neural ao ser exposta a um conjunto de dados consegue produzir conhecimento através do mesmo, e assim consegue reproduzir esse conhecimento em dados semelhantes àqueles a ela expostos.

2.1.1 O neurônio biológico

A capacidade cognitiva do ser humano está localizada no cérebro, onde as entidades básicas são os neurônios que, conectados em uma gigantesca rede, possibilitam a transmissão de informações criando assim a inteligência biológica. O ser humano na tentativa de mapear a estrutura e funcionamento do cérebro em um ambiente técnico, por exemplo combinando *hardware* e *software*, criou as redes

neurais artificiais, que buscam reproduzir a inteligência residente nos neurônios (MARCHIORI, 2006).

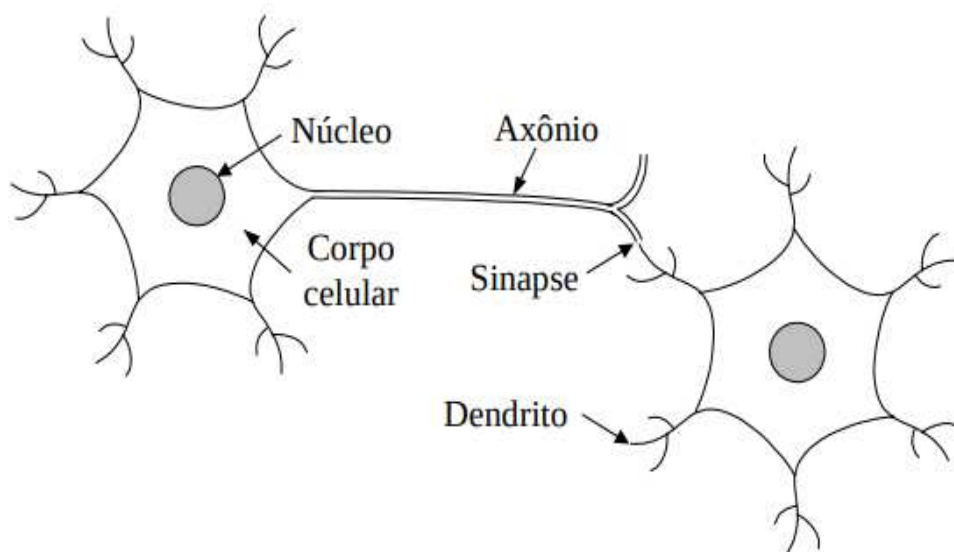
Ainda em Marchiori (2006), redes neurais vêm sendo utilizadas na mais diversas áreas, para soluções dos mais diversos tipos de problemas dentre eles: reconhecimento de padrões, diagnóstico e classificação, diagnósticos médicos, etc. Os quais têm relação direta com o objetivo desse trabalho.

Marchiori (2006), informa que as principais características do cérebro humano que as redes neurais tentam simular são:

- *Robustez e tolerância a falhas*: mesmo com a eliminação de neurônios, a capacidade geral neural não é afetada. Inclusive esse é um método comum chamado *dropout*, onde alguns neurônios são desativados propositalmente.
- *Capacidade de aprender*: o cérebro exposto à informações do exterior consegue aprender e gerar conhecimento.
- *Processamento de informação incorreta*: é possível que o raciocínio seja correto, mesmo quando as informações estão incompletas, apresentam ruído ou parcialmente contraditórias.
- *Paralelismo*: o número de neurônios ativos simultaneamente é enorme.

São cerca de 100 bilhões de neurônios processando as informações e nos dando a capacidade cognitiva. Eles possuem uma estrutura relativamente simples. São uma célula composta por núcleo e corpo celular (*sôma* do grego), onde reações químicas elétricas representam o processamento da informação. A informação parte do corpo celular realizada de impulsos elétricos, onde se propaga através do *axônio*. Os neurônios vizinhos recebem a informação através das inúmeras ramificações do *axônio*, as *sinapses*. As *sinapses* são as responsáveis pela ligação dos neurônios, elas ficam conectadas aos *dendritos* dos neurônios receptores. Ao ser excitada pelo impulso do *axônio*, a *sinapse* dispara uma substância química ao dendrito, realizando a conexão entre os neurônios. Conforme a excitação ou inibição, que a célula recebe de suas vizinhas, ela processa novamente a informação e a transmite adiante (MARCHIORI, 2006 *apud* SIMPSON, 1989; HAYKIN, 1994; MENDES FILHO & CARVALHO, 1997). O modelo simplificado de neurônio é mostrado na Figura 1.

Figura 1 Componentes de um neurônio biológico

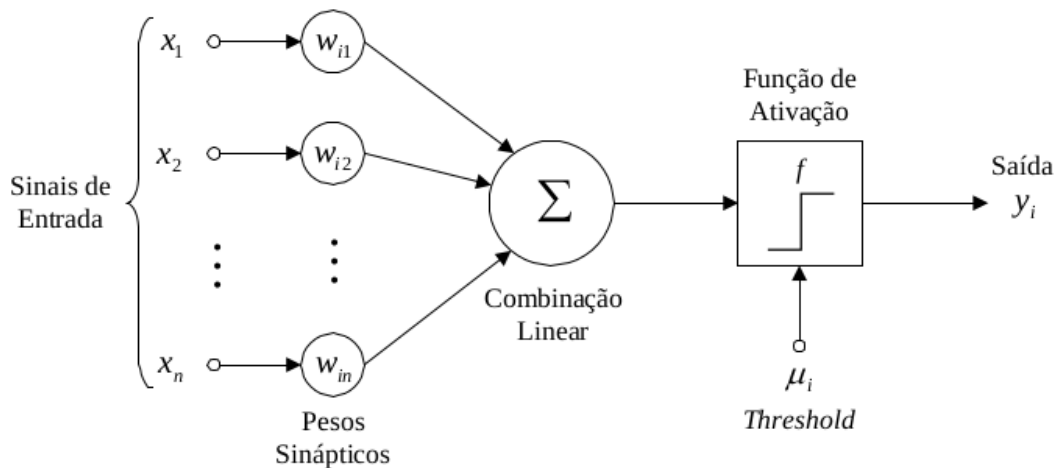


Fonte: Marchiori (2006)

2.1.2 O Neurônio Artificial

O neurônio artificial é o principal constituinte das RNAs, eles simulam os neurônios biológicos, e como tais, recebem estímulos de entrada, processa-os e retransmite aos demais neurônios conectados a ele. De acordo com Haykin e Engel (2007), o neurônio é a unidade fundamental de RNAs, formando a base de projetos das mesmas, sendo formados por três elementos básicos, os sinais de entrada, um somador e uma função de ativação. A Figura 2, representa a estrutura de um neurônio artificial.

Figura 2 Estrutura de um neurônio artificial



Fonte: Marchiori (2006)

O conjunto de entradas x_1 a x_n é conectado a um neurônio k , onde as entradas são multiplicadas por um peso sináptico w_{kj} na entrada da *sinapse* j . Os pesos podem ser tanto valores positivos quanto negativos. Os pesos sinápticos são os responsáveis pelo armazenamento do conhecimento que a rede adquire durante o treinamento. O *somador* é o elemento do neurônio responsável pela soma das entradas ponderadas pelos pesos. Essa operação constitui uma combinação linear. E por último temos a *função de ativação*. Ela limita a amplitude do sinal de saída do neurônio, e por isso também pode ser chamada de *função restritiva*. Comumente, a saída do neurônio, normalizada pela função de ativação, fica no intervalo fechado $[0, 1]$ ou $[-1, 1]$, no entanto há funções de ativação que não limitam superiormente o valor, tendo seus intervalos em $[-1, \infty]$ ou ainda $[0, \infty]$. Outro elemento é o *bias*, ele aumenta ou diminui a entrada da função de ativação, dependendo se é positivo ou negativo, respectivamente. É denotado por b_k (HAYKIN e ENGEL, 2007).

Ainda em Haykin e Engel (2007), denotamos um neurônio, matematicamente, como duas funções:

$$u_k = \sum_{j=1}^n w_{kj} x_j \quad (1)$$

e

$$y_k = \varphi(u_k + b_k) \quad (2)$$

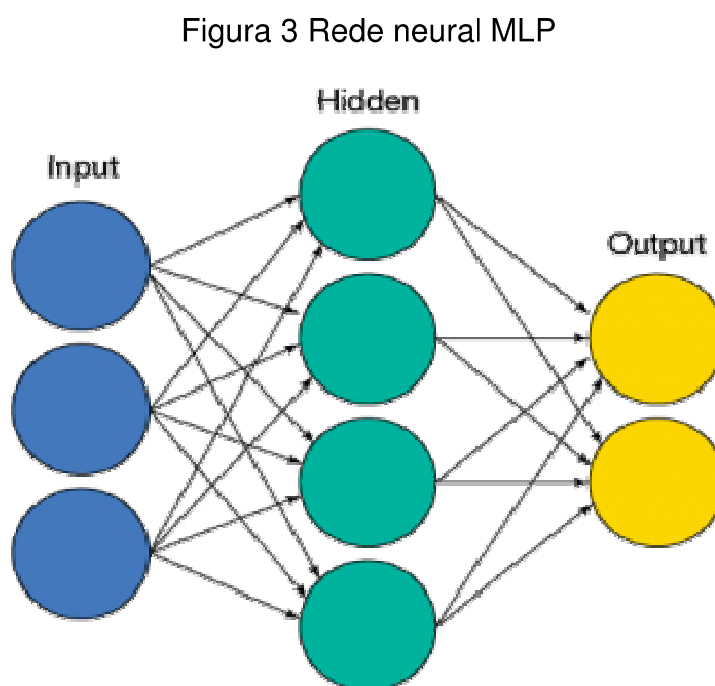
onde x_1 a x_n são os sinais de entrada, w_{k1} a w_{kn} são os pesos sinápticos, u_k é a saída da combinação linear, b_k é o bias, φ é a função de ativação, e y_k é o sinal de saída do neurônio.

2.1.3 Redes Feedforward

Redes neurais são organizadas em camadas de neurônios, na sua forma mais simples tem-se uma camada de entrada e uma camada de saída. A alimentação dos dados na rede ocorre da camada de entrada para a camada de saída, mas nunca vice-versa, ou seja, ela é alimentada somente em um sentido, assim é dito que ela é alimentada adiante (*feedforward*) (HAYKIN e ENGEL, 2007).

Um modelo de rede neural *feedforward* muito conhecido é o *Multilayer Perceptron* (MLP), que apresenta várias camadas. Cada camada fica totalmente conectada à camada adiante, ou seja, cada neurônio da camada está conectado a todos os neurônios da camada imediatamente à frente, a esse tipo de camadas dá-se o nome de totalmente conectada (*fully connected*), ou camada densa.

Cada camada recebe um nome de acordo com sua localização na rede. A primeira camada de uma rede neural é chamada camada de entrada, ela é a responsável pela entrada dos dados na rede. A última camada da rede é chamada de camada de saída, nela tem-se o resultado do processo de *feedforward*, ou seja, a predição que a rede realiza. As camadas entre estas duas mencionadas, são chamadas de camadas intermediárias ou ocultas, ou *hidden layers*. A Figura 3, mostra uma rede com três camadas, sendo: camada de entrada, camada oculta e a camada de saída.



Fonte: iMasters¹

2.1.4 Funções de ativação

A função de ativação delimita um limiar para a saída de um neurônio, é também responsável pela não-linearidade da rede neural.

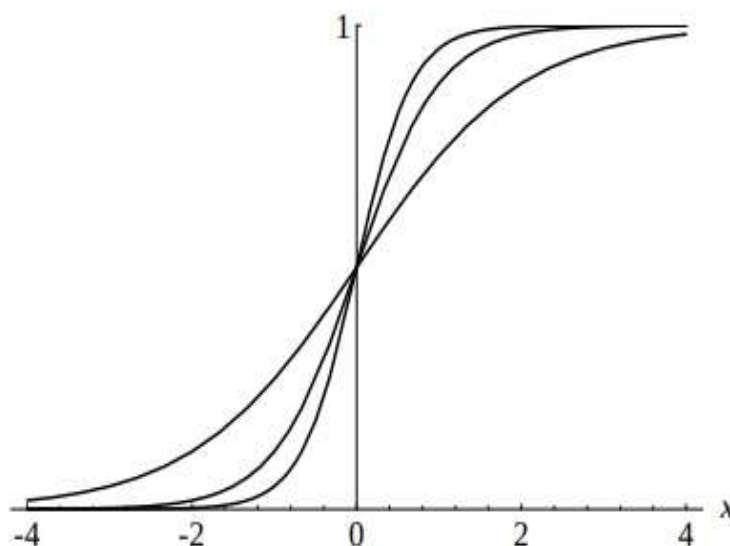
¹ iMasters: <https://imasters.com.br/back-end/desenvolvimento-de-solucoes-cognitivas-iot-para-deteccao-de-anomalias-atraves-da-aprendizagem-profunda-parte-1>

Função sigmóide: é uma das funções de ativação mais usadas em redes neurais. Sua forma é de S. Ela é estritamente crescente e exibe um balanceamento adequado entre comportamento linear e não-linear. Um exemplo de função sigmóide é a função logística, definida como

$$f(x) = \frac{1}{1+e^{-ax}}, \quad (3)$$

onde a define a inclinação da função, variando-se a , obtêm-se funções sigmóides com diferentes inclinações (HAYKIN e ENGEL, 2007). A Figura 4, representa a função sigmóide com diferentes inclinações.

Figura 4 Gráfico da função sigmóide, para valores de $a=1$, $a=2$ e $a = 4$



Fonte: ROJAS (1996)

Função softmax: é uma das funções mais utilizadas em problemas de classificação, pois a mesma tem como valor de ativação uma probabilidade estatística das classes de saída, ou seja, qual a probabilidade de uma determinada classe ser a correta. Diferentemente das funções sigmóide que resulta valores do quanto a rede “acha” que uma saída é a correta, a *softmax* dá a probabilidade, assim a soma de todos os valores de ativação vão sempre resultar no valor 1.0.

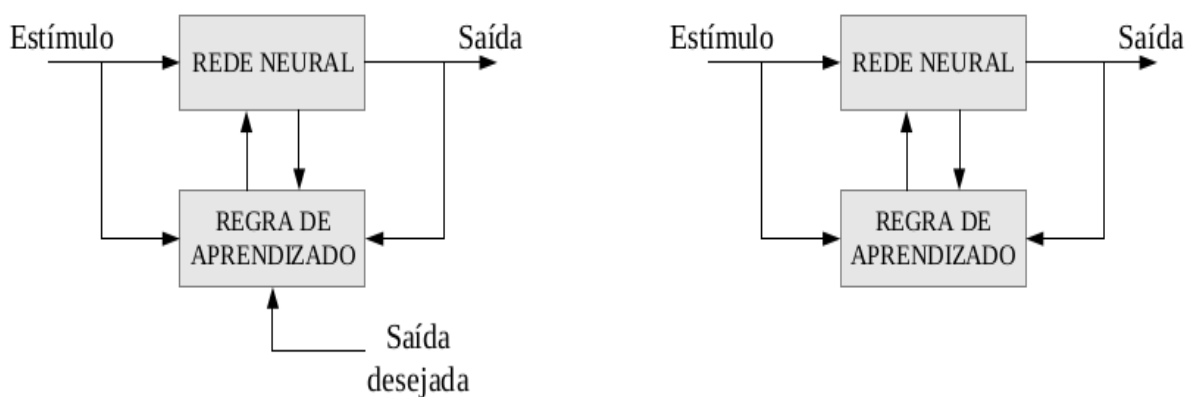
Um fator importante no treinamento das redes neurais, é a forma como ela se relaciona ao ambiente, ou seja, como ocorre o processo de aprendizagem da rede.

Existem três maneiras diferentes, principais, de treinar uma rede neural (HAYKIN e ENGEL, 2007)

- *Treinamento Supervisionado*: o conhecimento é adquirido através de dados de entrada, levando-se em consideração a saída desejada, ou seja, um dado é apresentado e a saída esperada é mostrada a rede, e através de um processo de ajuste dos pesos, a rede aproxima sua saída à esperada.
- *Treinamento Não-supervisionado*: nesse tipo de treinamento apenas os dados de entradas são usados para ajuste dos pesos. Não há um professor informando o resultado esperado, como é o caso do supervisionado. Como forma de ajustar os pesos corretamente, métricas ou regras são definidas para informar o quanto a rede está indo bem ou não.
- *Treinamento Híbrido*: esse faz uma junção dos dois métodos citados, parte do treinamento ocorre de maneira supervisionada e parte de maneira não-supervisionada.

Nesse trabalho, o método utilizado é o treinamento supervisionado. Onde as entradas são um conjunto de imagens, e a saída é uma lista de valores indicando o quanto a rede acha que a imagem é de um tipo ou de outro, ou seja é cárie ou não. A Figura 5 ilustra os dois métodos, supervisionado e não-supervisionado, respectivamente.

Figura 5 Modelo de treinamento supervisionado à esquerda e não-supervisionado à direita



Fonte: Marchiori (2006)

2.1.5 Backpropagation

A rede neural é capaz de resolver muitas classes de problemas, no entanto, para que isso ocorra é necessário que os pesos corretos sejam encontrados. Para tal usa-se o algoritmo de *backpropagation*. Esse método busca minimizar a função de perda através da atualização dos pesos, utilizando-se para isso o método do gradiente descendente. Esse método requer o cálculo dos gradientes da função de perda a cada iteração, por isso é necessário manter a continuidade e diferenciabilidade da função de erro (ROJAS, 1996).

O *backpropagation* é uma técnica utilizada para implementar a descida de gradiente nas redes do tipo *feedforward*. Ela deve calcular de forma eficiente as derivadas parciais de uma função $F(w, x)$ realizada pela rede em relação aos pesos (w) e um dado conjunto de dados como entrada (x) (HAYKIN e ENGEL, 2007).

O cálculo função de perda é essencial para que o erro da rede seja obtido. Para que com a minimização da função de perda, os pesos sejam reajustados através da retro-propagação. O erro é dado pela equação:

$$E = \frac{1}{2} \sum_{i=1}^p \|o_i - t_i\|^2, \quad (4)$$

onde o são as saídas da rede, t é a saída desejada e p é a quantidade de pares $\{(x_1, t_1), \dots, (x_p, t_p)\}$ dos dados de treino. O algoritmo de *backpropagation* é usado para encontrar os mínimos locais da função de perda, e então atualizar os pesos, uma vez que esses são inicializados de forma aleatória (ROJAS, 1996).

O erro quadrático E pode ser minimizado usando um processo iterativo de gradiente descendente, para o qual é necessário calcular o gradiente, dado por

$$\nabla E = \left(\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots, \frac{\partial E}{\partial w_l} \right) \quad (5)$$

Cada peso é atualizado usando a equação de incremento:

$$\Delta w_i = -\gamma \left(\frac{\partial E}{\partial w_i} \right) \text{ for } i = 1, \dots, l \quad (6)$$

onde γ representa a constante de aprendizado, ou taxa de aprendizado. É um parâmetro de proporcionalidade que define o comprimento do passo de cada iteração na direção negativa do gradiente descendente. Valores muito altos podem fazer a rede nunca convergir, ou seja, chegar a um erro aceitável; já valores muito baixos podem acarretar que a rede demore muito a convergir. O mais comum é usar potências negativas em 10, como 10^{-1} , 10^{-2} , 10^{-3} e 10^{-4} . Embora a taxa de aprendizado seja geralmente constante, alguns métodos usam uma taxa variável.

2.2 Rede neural convolucional (CNN)

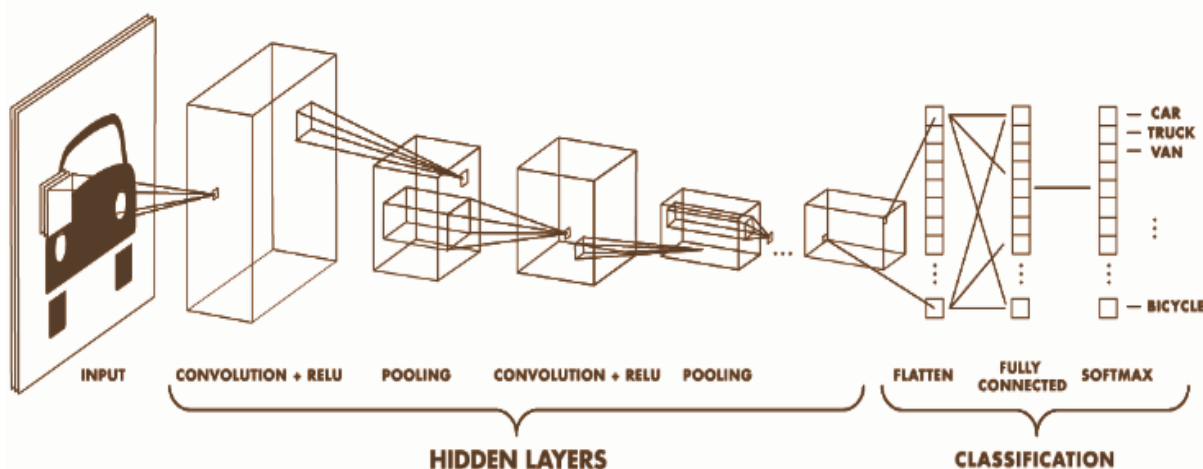
Uma rede neural convolucional é um perceptron de múltiplas camadas projetado para trabalhar com formas bidimensionais, imagens, com alta invariância referente a translação, escalonamento, inclinação e outras distorções (HAYKIN e ENGEL, 2007).

A rede deve ser treinada de forma supervisionada e sua estrutura inclui as seguintes restrições (HAYKIN e ENGEL, 2007, et. al LECUN e BENGIO, 1995):

- *Extração de características*: cada neurônio é forçado a extrair características locais, uma vez que suas entradas são oriundas de um campo receptivo local na camada anterior. A localização exata da característica vai perdendo importância à medida que avança na rede, mas a posição em relação às outras características deve ser mantida.
- *Mapa de características*: cada camada convolucional da rede é composta por múltiplos mapas de características.
- *Sub Amostragem (downsampling)*: as camadas convolutivas devem ser seguidas por uma camada que calcula a média local e faz sub amostragem, reduzindo a resolução do mapa de características.

Nas arquiteturas atuais é comum ter muitas camadas convolutivas antes de uma camada de *downsampling*, embora nas primeiras camadas seja interessante utilizar o *downsampling*, para redução do consumo de recursos da máquina. A Figura 6 mostra a estrutura de uma rede neural convolucional.

Figura 6 Esquema de rede neural convolucional



Fonte: Freecodecamp²

A primeira camada é onde as entradas serão fornecidas à rede. A segunda camada é a primeira convolutiva onde ocorrerá a aplicação de filtros, inicialmente aleatórios, através de convolução. A terceira camada é a camada de *pooling* (uma forma de *downsampling*). A quarta camada é novamente de convolução e segue essa alternância de convolução e *pooling*.

Para a tarefa de classificação, não bastam camadas convolutivas, é necessário uma rede neural do tipo MLP ao final, como é possível notar na imagem, na parte *classification*. Nessa parte o último mapa de características vai ser interpretado e a predição é feita.

2.2.1 Convolução

A convolução é o somatório do produto de um *pixel* e seus vizinhos por um *kernel*, ou máscara, a fim de gerar um outro *pixel* de uma nova imagem. Para isso é necessário entender como as imagens são interpretadas por computadores.

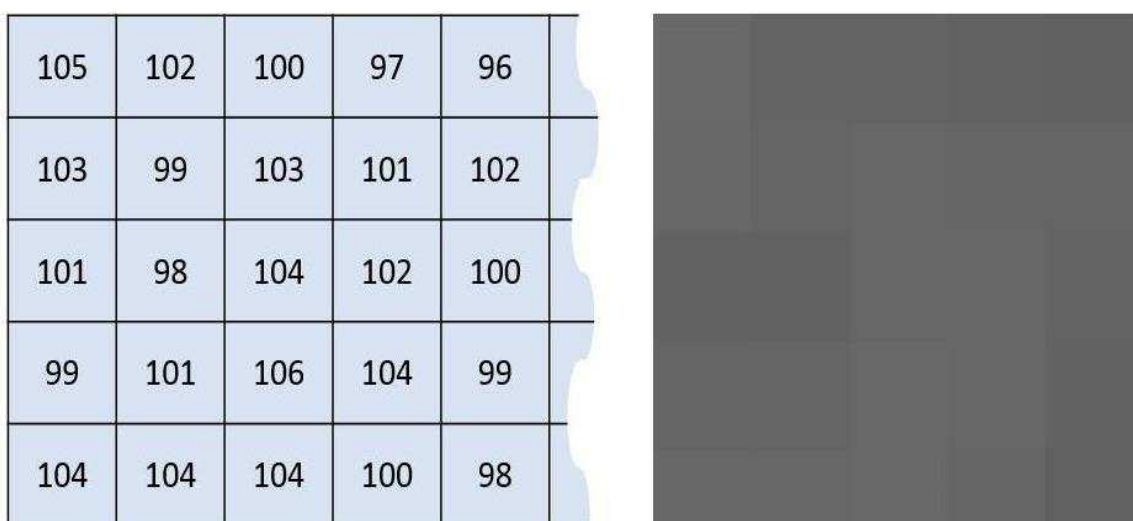
Uma imagem digital nada mais é que uma matriz bidimensional ou tridimensional de valores inteiros entre 0 e 255, o valor 0 representa a ausência de cor, ou seja, a cor preta; o valor 255 representa a cor branca.

A matriz que forma uma imagem difere caso ela seja em escala de cinza ou colorida. Imagens em escala de cinza tem apenas duas dimensões, linhas e

² Freecodecamp: <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>

colunas. Já as imagens coloridas têm três dimensões, sendo linhas, colunas, e profundidade. Cada valor em uma imagem representa um *pixel*. Nas imagens coloridas os *pixels* são formados por três valores, o primeiro valor representa a quantidade de vermelho (R do inglês *Red*), o segundo valor o verde (G do inglês *Green*) e o terceiro é o azul (B do inglês *Blue*). Esse é o padrão de imagens em aparelhos digitais, o RGB. Na Figura 7 temos uma representação de uma pequena região de uma imagem. Do lado esquerdo temos os valores dos *pixels* e do lado direito a imagem.

Figura 7 Imagem e sua representação digital



Fonte: Machinelearninguru³

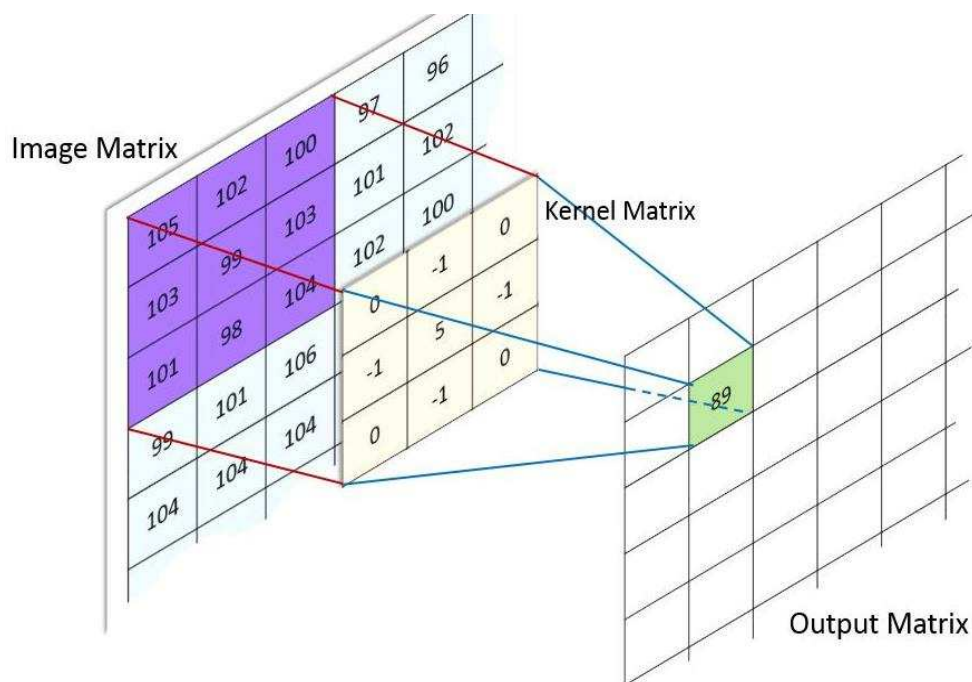
O processo de convolução sobre uma imagem digital produz outra imagem. Uma pequena matriz (*Kernel*) é multiplicada por cada *pixel* da imagem dando origem aos *pixels* da nova imagem.

Na Figura 8 temos uma representação desse processo. O *kernel* (3x3) é multiplicado por cada *pixel* da imagem original. Cada valor no *kernel* é multiplicado pelos *pixels* vizinhos a um *pixel* X em questão, no caso do exemplo, o *pixel* em questão é o de valor 99, no centro da região roxa.

³ Machinelearninguru:

http://machinelearninguru.com/computer_vision/basics/convolution/image_convolution_1.html

Figura 8 Processo de convolução

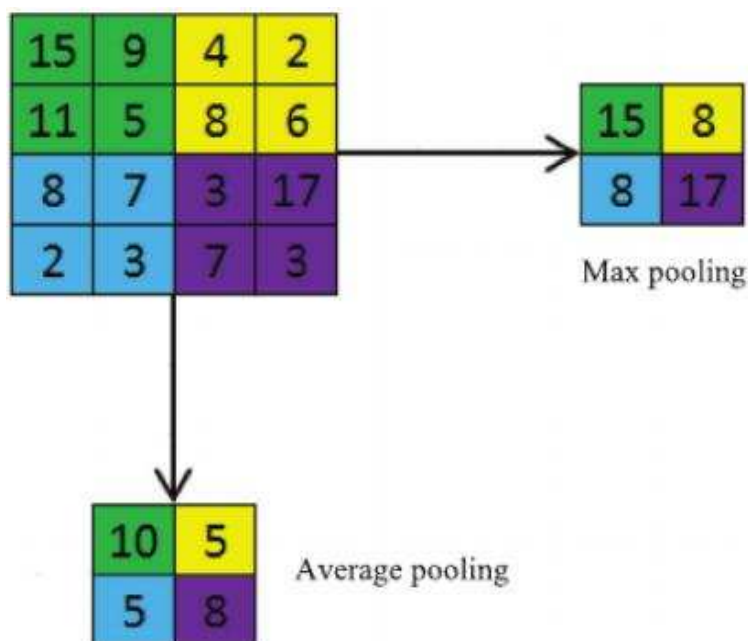


Fonte: Machinelearninguru³

2.2.2 Downsampling

O processo de *downsampling* consiste em reduzir a dimensão dos mapas de características. O processo mais comum de *downsampling* é o *pooling*. O método mais comum de *pooling* é o *maxpooling*. Esse processo aplicado em uma imagem, ou mapa de características, reduz o tamanho a dimensão pela metade, mas mantém as características da imagem. O processo de *pooling* pode ser realizado também através de outros métodos diferentes do *maxpooling*, como o *average pooling* e o *minpooling*. Na Figura 9 nota-se a diferença entre o *maxpooling* e o *average pooling*. O primeiro seleciona o maior valor dada uma janela 2x2 da imagem, já o segundo calcula a média da janela.

Figura 9 Diferença entre maxpooling e average pooling



Fonte: Rawat e Wang (2017)

Antes do processo de *pooling* a imagem original tinha 4x4 *pixels*, após o *pooling* uma nova imagem é gerada a partir da original, a qual ficou com 2x2 *pixels*, ou seja, metade do tamanho original. Contudo as características da imagem foram preservadas.

2.2.3 Arquitetura de uma CNN

Como já foi possível notar, CNNs são projetados utilizando três tipos básicos de camadas, as camadas de convolução, as camadas de *pooling* e as camadas altamente conectadas.

A arquitetura mais comum de CNN, atualmente, é uma pilha de CONV (*convolutional layer*) e RELU, seguidas de camadas de POOL (*pooling*), e repetido esse processo até finalmente juntar-se com as *fully-connected layers*, onde a saída da rede é gerada (KARPATHY, 2015).

2.3 Algoritmos Genéticos (AG)

Nas décadas de 1950 e 1960 vários cientistas da computação estudavam, independentemente, sistemas evolucionários. Eles esperavam usar estes sistemas como otimizadores para ferramentas de problemas em engenharia. A ideia por trás desses algoritmos é evoluir uma população de soluções candidatas a resolução de um dado problema, usando operadores inspirados por variações genéticas naturais e seleção natural (MITCHELL, 1996).

Métodos baseados em computação evolutiva, como os algoritmos genéticos, são inspirados na teoria de evolução natural, apresentada por Darwin (1859). A grande vantagem dos algoritmos genéticos está na capacidade de resolução de problemas de alta complexidade através de uma simples descrição matemática da solução almejada. Além do fato do algoritmo ser genérico, e portanto facilmente adaptável ao problema, uma vez que os passos a serem seguidos são os mesmos (FRANCISCO, 2013).

A evolução biológica é a base de inspiração para a solução de problemas onde técnicas computacionais tradicionais seriam ineficazes ou não aplicáveis. A evolução é, na verdade, um método de busca entre um enorme número de possibilidades de soluções. Na biologia, o enorme conjunto de possibilidades é o conjunto de sequências genéticas possíveis, e as soluções desejadas são organismos altamente aptos, ou seja, organismos que conseguem sobreviver e reproduzir-se em seus ambientes (MITCHELL, 1996).

A computação evolutiva parte da idéia que dada uma população de indivíduos, forçados pela pressão do meio ambiente onde estão situados, são pressionados a uma evolução através da seleção natural - os indivíduos menos aptos são eliminados, enquanto os mais aptos se reproduzem e passam seus genes adiante, assim transmitindo suas melhores características aos seus descendentes.

Em sistemas evolutivos algumas das terminologias aplicadas fazem analogia aos termos biológicos naturais. Embora as estruturas computacionais sejam muito mais simples que as biológicas, porém a simulação desses elementos biológicos tem trazido grandes resultados para os problemas de otimização. Os principais termos adotados do contexto natural são explicados a seguir.

- *Cromossomo*: são tipicamente representados em forma de cadeias de *bits*, as quais podem ter dois possíveis valores: 0 ou 1. Cada cromossomo pode ser considerado como uma solução no espaço de pesquisa de soluções candidatas (MITCHELL, 1996).
- *Crossover (recombinação)*: é o processo de cruzamento dos indivíduos ou cromossomos, que a partir de dois cromossomos um terceiro é gerado, e partilha das características dos outros dois. A *recombinação* une os genes de dois ou mais genótipos para gerar um ou dois filhos. Este operador é estocástico, assim a escolha de quais genes de cada pai a serem escolhidos para recombinação é aleatória (MITCHELL, 1996).
- *Genes*: são a unidade fundamental dos cromossomos, geralmente são representados como 0 ou 1.
- *Mutação*: assim como recombinação, produz novos filhos a parte de um indivíduo. A mutação é realizada através da alteração aleatória dos genes de um cromossomo.
- *Seleção*: é o processo de escolha dos indivíduos com base na sua adaptação ao meio, ou seja, sua função de *fitness*. O papel da seleção é tornar os melhores indivíduos pais das próximas gerações (ALMEIDA, 2003). Dessa forma perpetuando os melhores genes.

2.3.1 Função de *fitness*

A função de *fitness* ou aptidão é utilizada no processo de seleção dos indivíduos. Ela deve indicar a adaptação do indivíduo na população e no meio, ou seja, ela é uma métrica que quantifica a adaptabilidade do indivíduo (ALMEIDA, 2003).

Nesta pesquisa a adaptabilidade do indivíduo é medida através da acurácia da rede. Assim a cada geração os indivíduos com as melhores acurácias são mantidos e cruzados.

2.3.2 Critério de parada

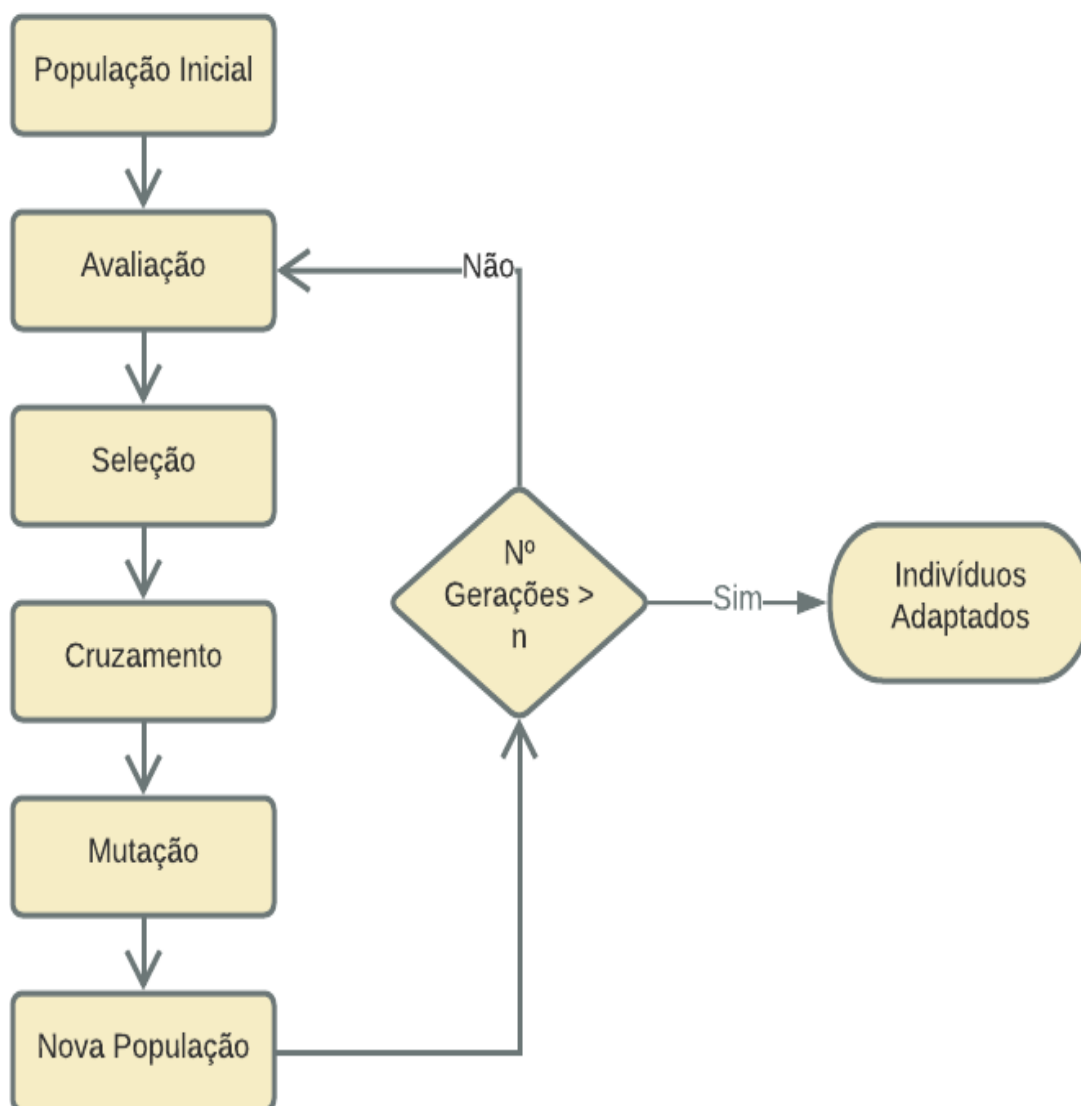
Outro ponto importante para levar-se em consideração é o critério de parada. Para isso poderia ser utilizado o valor ótimo da função de *fitness*, contudo, os AG's são estocásticos, então essa condição pode nunca ser alcançada. Assim o algoritmo

pode nunca parar. Para contornar isso alguns critérios podem ser utilizados (ALMEIDA, 2003):

- tempo máximo de execução do AG;
- quantidade de avaliações da função de aptidão atingir um limite, ou seja, a quantidade de cromossomos já gerados;
- quando a adaptação dos indivíduos melhora muito em pouco tempo;
- quando a diversidade da população atinge um determinado limite. Por exemplo, os indivíduos mostram-se muito similares.

A Figura 10 mostra o diagrama de um processo evolutivo realizado por algoritmos genéticos.

Figura 10 Diagrama do processo evolutivo de algoritmos genéticos



Fonte: Próprio autor

2.4 Tecnologias

A implementação do modelo proposto foi realizada utilizando o *framework* keras⁴, o qual fornece uma grande API de alto nível, para se trabalhar com aprendizado de máquinas, por exemplo: redes neurais e regressão linear.

O algoritmo genético utilizado nesse trabalho foi adaptado do repositório *neural-network-genetic-algorithm*⁵. Para a utilização com CNNs foram feitas adaptações, pois o mesmo só funcionava para RNAs.

A linguagem de programação utilizada tanto no keras como no AG é a linguagem Python⁶. O algoritmo⁷ desenvolvido pelo autor encontra-se disponível em um repositório público no site Github.com.

Nesse capítulo foram apresentados os conceitos essenciais para a compreensão da metodologia e das técnicas empregadas no decorrer da pesquisa. O próximo capítulo abordar as etapas realizadas no desenvolvimento do modelo proposto.

⁴ Keras: <https://keras.io/>

⁵ Algoritmo Genético: <https://github.com/harvitronix/neural-network-genetic-algorithm>

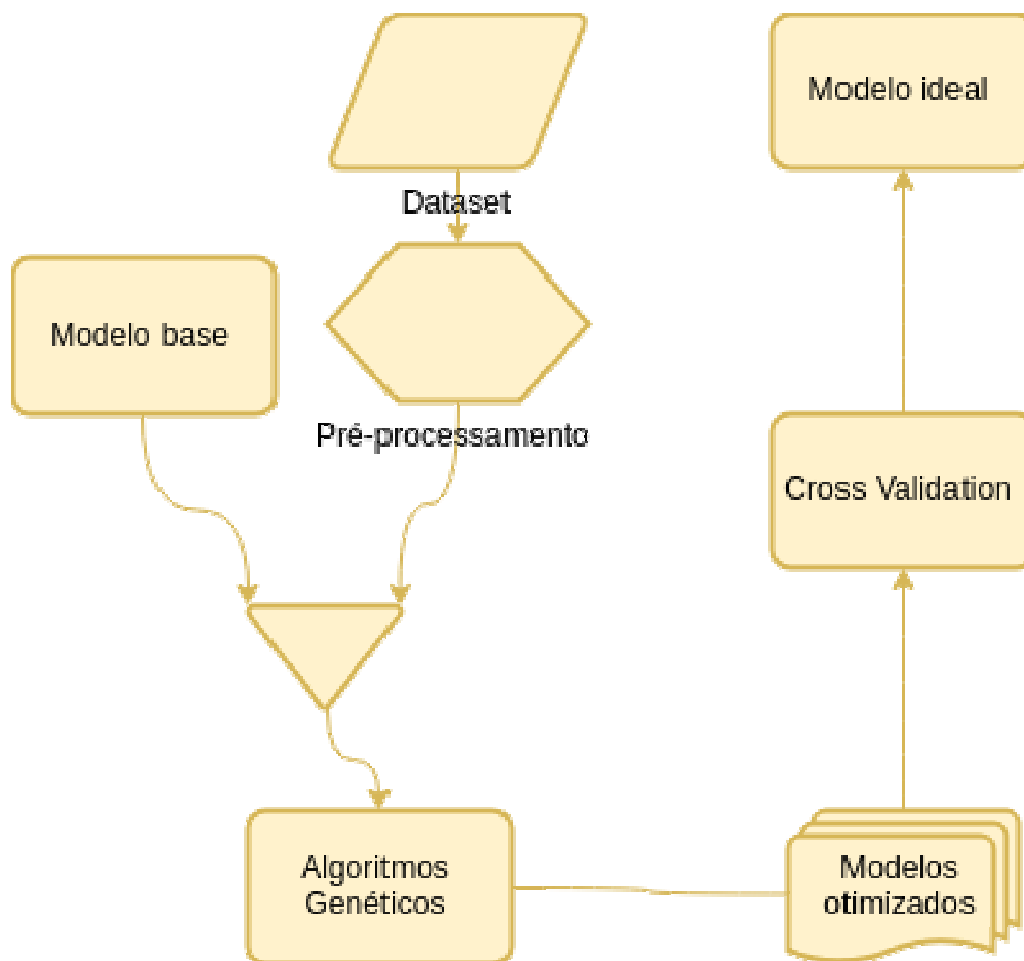
⁶ Python: <https://www.python.org/>

⁷ Algoritmo Genético adaptado: <https://github.com/rromildosf/ga4ann>

3 DESENVOLVIMENTO

Neste capítulo serão apresentadas as etapas que englobam o desenvolvimento da arquitetura de CNN proposta por esse trabalho. A Figura 11 mostra o processo de escolha do melhor modelo.

Figura 11 Diagrama do processo de escolha do melhor modelo



Fonte: Próprio autor

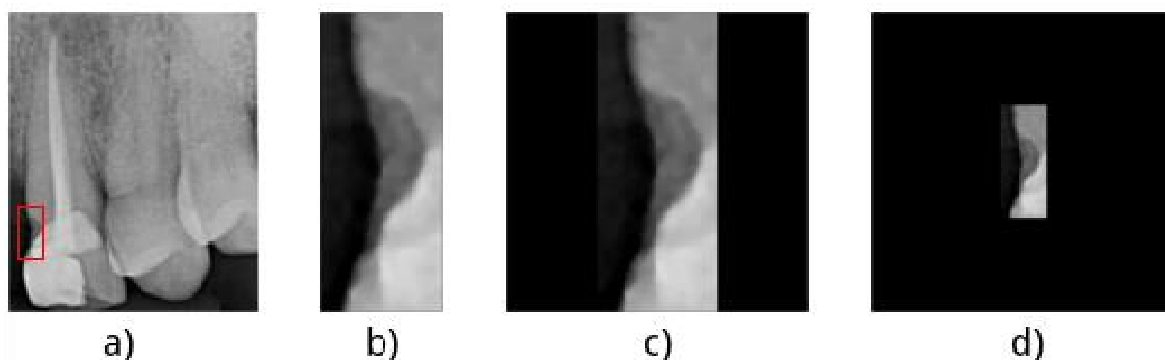
A primeira parte do processo foi a obtenção das amostras, através de um pré-processamento. Um modelo básico foi definido, e o AG completou sua estrutura, após isso treinou a população de cada geração com as amostras. Ao final do treinamento evolucionário no AG, os cinco modelos com as melhores acurácias foram novamente treinados, mas dessa vez utilizando o processo de *cross validation*. E o melhor modelo foi selecionado, para a solução do problema inicialmente levantado.

Nas seções seguintes cada um destes processos será descrito com detalhes.

3.1 Preparação do *dataset*

Uma vez que a rede neural precisa de um tamanho predefinido das imagens que são utilizadas para treinamento, e as amostras obtidas têm tamanhos distintos, elas devem ser normalizadas para um tamanho comum. Existem duas formas de se fazer isso. A primeira é centralizar a amostra original e preencher com a cor preta o espaço que falta para preencher todo o novo tamanho, esse processo é chamado *padding*. A segunda é expandir a amostra para o novo tamanho. A Figura 12, exibe a imagem original (a), a amostra extraída (b), amostra expandida ao tamanho de 256x256 *pixels* e d) amostra com tamanho original mantido, mas expandida ao tamanho de 256x256 *pixels* usando *padding*.

Figura 12 Pré-processamento do dataset



a) imagem original da radiografia com marcação na região da cárie; b) amostra extraída; c) amostra expandida; d) amostra mantido o tamanho original mais adição de *padding*.

Fonte: Próprio autor

Os dois conjuntos de amostras passaram pelos processos de expansão e adição de *padding*, gerando assim quatro conjuntos de amostras, para determinar qual dos dois métodos é mais adequado ao problema.

Outro processo importante pelo qual as imagens foram submetidas, foi o processo de *data augmentation* (aumento de dados). Segundo Virdee (2018), *data augmentation* consiste em aplicar transformações nas imagens, como rotação, espelhamento, ruído, correção de contraste, ajuste de brilho, realce de bordas, distorções entre outras. No intuito de gerar novas imagens, aumentando um *dataset* com tamanho limitado. Esse é um método bastante útil, pois a mesma imagem passa por várias transformações permitindo que a mesma seja interpretada de várias formas, mas sem perder suas características, tal processo melhora a

qualidade da generalização da rede neural. A Tabela 1 mostra a quantidade de amostras com e sem *data augmentation*, e também seus tipos.

Tabela 1 Conjuntos de amostras

Tipo de amostra	Amostras originais	Com <i>data augmentation</i>
Com cárie + <i>padding</i>	349	3490
Com cárie + expansão	349	3490
Sem cárie + <i>padding</i>	348	3480
Sem cárie + expansão	348	3480

Fonte: Próprio Autor

3.2 Escolha dos hiperparâmetros

Nesta etapa um modelo base de CNN foi desenvolvido com um mínimo de hiperparâmetros pré-definidos, uma vez que os demais devem ser escolhidos pelo algoritmo genético, e coleções de hiperparâmetros foram definidas, as quais passaram pelo processo de evolução com o uso de algoritmos genéticos, e através do processo evolutivo, os melhores foram escolhidos. A seguir uma descrição dos hiperparâmetros utilizados.

- ***cnn_nb_layers***: coleção de valores que define quantas camadas convolutivas a rede receberá.
- ***ann_nb_layers***: coleção de valores que define a quantidade de camadas densas.
- ***cnn_layers***: coleção de valores que define a quantidade de filtros de cada camada convolutiva, cada uma recebe um valor distinto.
- ***ann_layers***: coleção de valores que define a quantidade de neurônios de cada camada densa, cada uma recebe um valor distinto.
- ***cnn_activations***: coleção de funções de ativação a serem usadas nas camadas de filtros (uma única função para todas as camadas).
- ***ann_activations***: coleção de funções de ativação a serem usadas nas camadas de densas (uma única função para todas as camadas).

- **optimizers**: coleção de otimizadores, apenas um é usado em toda a rede.
- **dropouts**: coleção de valores. Definem a quantidade de camadas de *dropout*, sendo *dropouts* de 50%.
- **poolings**: coleção de valores. Definem a quantidade de camadas de *pooling* na rede.

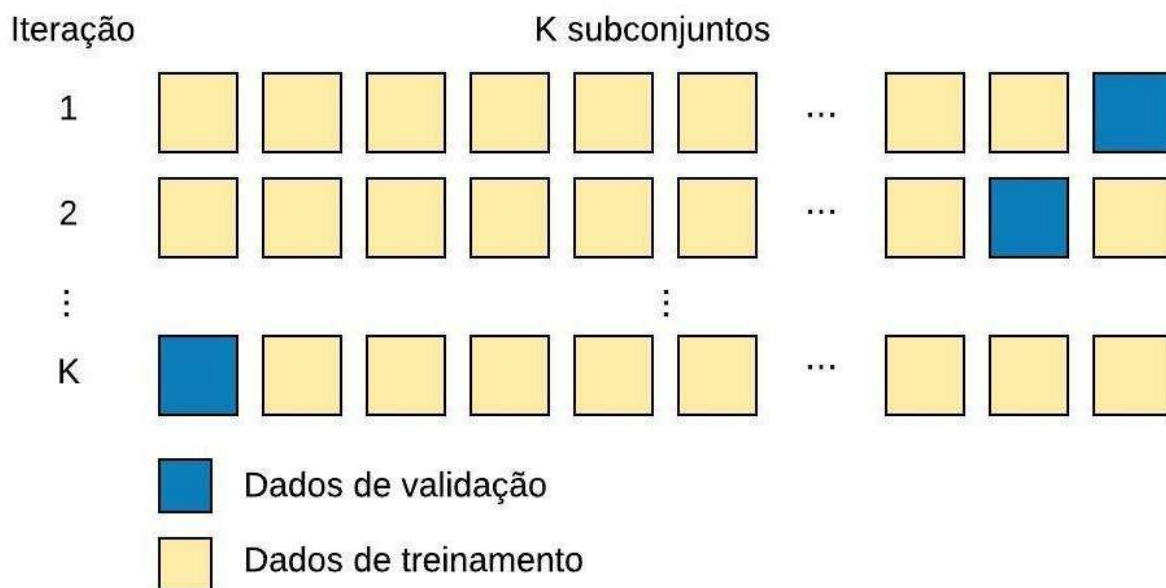
Durante o processo evolutivo esses hiperparâmetros são escolhidos, em princípio aleatoriamente, e usados para gerar a CNN. A cada nova geração os melhores indivíduos, ou seja hiperparâmetros, são mantidos e os piores são descartados.

3.3 Validação cruzada

Validação cruzada é um método estatístico de avaliação e comparação de algoritmos de aprendizagem, é realizado através da divisão dos dados em dois subconjuntos, um utilizado para treinar e o outro utilizado para a validação do modelo. Os subconjuntos precisam passar pelo processo de cruzamento sucessivas vezes. A maneira mais comum de validação cruzada é a *K-fold cross validation*, onde os dados são divididos em K segmentos de igual tamanho. Em seguida K iterações de treinamento e validação são realizadas, em cada uma dessas, um segmento diferente é utilizado para validação, enquanto $K-1$ segmentos são usados para treinamento (REFAEILZADEH, TANG e LIU, 2009).

Em cada iteração de treinamento e validação, é armazenada a acurácia do modelo e ao final das K iterações uma média aritmética é realizada com esses valores, e essa se torna a acurácia do modelo para aquele conjunto de dados. Dessa maneira todos os dados participam dos processos tanto de treinamento quanto de validação. A validação cruzada é comumente utilizada para verificar a capacidade de generalização de um dado algoritmo. A Figura 13, apresenta o processo:

Figura 13 Representação dos dados no processo de validação cruzada



Fonte: Próprio autor

Como é possível observar, na primeira iteração de treinamento os nove primeiros segmentos serão utilizados para treinamento do modelo, e o último para validação. Já na segunda iteração os oito primeiros mais o décimo serão os dados de treinamento, enquanto o nono será utilizado na validação. Isso possibilita medir a precisão para todo o conjunto de dados, em vez de um único subconjunto do mesmo.

3.4 Aplicação dos algoritmo genético

Os algoritmos genéticos foram utilizados nesse trabalho como um otimizador para os hiperparâmetros da rede neural. Os hiperparâmetros a serem otimizados pelos AGs foram divididos em duas categorias; i) os necessários as camadas de convolução e *pooling*, sendo; função de ativação, quantidade de camadas de convolutivas e de *pooling*, quantidade de filtros em cada camada convolutiva; ii) os relacionados às camadas densas (totalmente conectadas), sendo; função de perda, função de ativação, otimizador, quantidade de camadas densas, quantidade de neurônios em cada camada e quantidade de camadas de *dropout*.

No AG foram utilizadas 30 gerações, cada uma delas contendo uma população de 30 indivíduos, ou seja, 30 redes neurais foram treinadas e avaliadas

em cada geração. No início do treinamento evolutivo, a população foi gerada inteiramente aleatória. O valor 30 foi utilizado, pois um valor muito maior demandaria muito tempo, como são 30 gerações e 30 indivíduos por geração, foram avaliadas 900 redes neurais.

Ao final do treinamento da primeira geração os indivíduos passam pelo processo de seleção. Para que isso ocorra é necessário avaliar a adaptação (*fitness*) de cada indivíduo, que é mensurada através da acurácia da rede nos dados de teste. Em seguida 50% dos mais adaptados são selecionados, enquanto que 50% são descartados. Foi utilizado metade dos indivíduos, pois cada par deles gera um filho, e como 50% mantém-se na próxima geração, temos um total de 30 indivíduos novamente. Os selecionados passam, então, pelo processo de cruzamento, onde têm seus genes (hiperparâmetros) cruzados. E por fim temos a mutação dos genes, onde pequenas mudanças são inseridas. Juntam-se os indivíduos mais adaptados e seus filhos e para compor a nova população.

Nesse capítulo foram abordados os passos do desenvolvimento e utilização dos algoritmos utilizados durante a pesquisa, assim preparação do conjunto de amostras utilizadas para treinamentos das populações de indivíduos (possíveis soluções) no AG. No próximo capítulo são apresentados e discutidos os resultados obtidos.

4 RESULTADOS

O *dataset* obtido foi dividido em dois conjuntos de amostras. Cada conjunto foi utilizado para treinamento das populações de cada geração obtidas com o uso dos algoritmos genéticos e o aprendizado mostrou-se bastante distinto, implicando que o pré-processamento correto dos dados é de vital importância para o treinamento adequado de uma arquitetura de rede neural.

No primeiro conjunto de amostras foi mantido o tamanho original da amostra retirada das imagens radiográficas originais e adicionados *pixels* com valor zero, ou seja, cor preta. Os valores da função de perda e da acurácia são mostrados a seguir, para o treinamento simples, em um primeiro momento, ou seja, sem usar *cross validation*. O gráfico da Figura 14 mostra os valores da acurácia nos dados de validação ao longo de 26 épocas de treinamento.

Figura 14 Gráfico da acurácia nos dados de validação



Fonte: Próprio autor

A reta de cor laranja escura mostra os resultados após um processo de suavização dos resultados. Já a reta mais clara são os resultados exatos. Como se pode observar o melhor resultado ficou em torno de 81% no *epoch* (iteração de treinamento) 17. A maior parte dos resultados, observando o gráfico, ficaram entre 74% e 78% a partir do *epoch* 17.

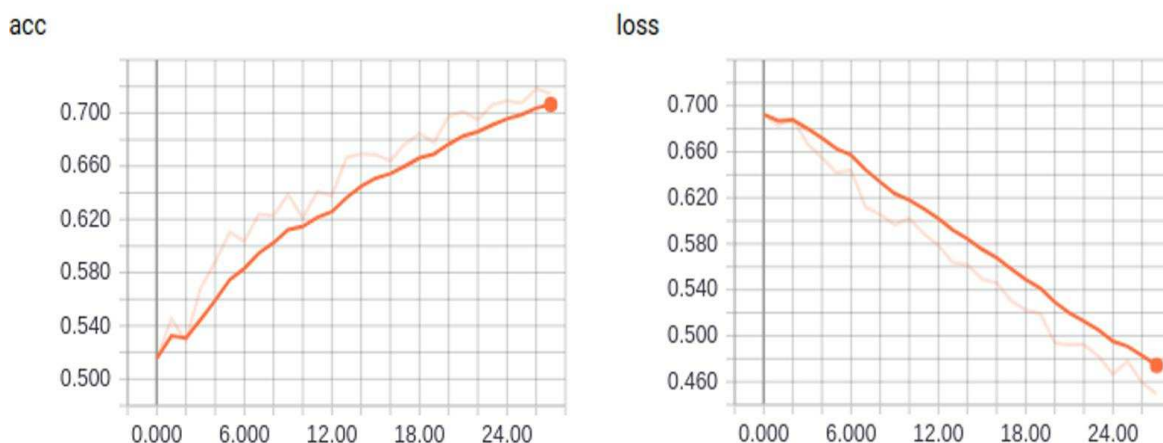
Figura 15 Gráfico da função de perda nos dados de validação



Fonte: Próprio autor

O gráfico da Figura 15, por outro lado, mostra os valores da função de custo para os dados de validação ao longo das 26 épocas de treinamento. Como é esperado, a função apresenta valores cada vez menores. Os melhores valores apresentados ficaram próximos de 0.5.

Figura 16 Valores da acurácia (esquerda) e da função de perda (direita), para os dados de treinamento

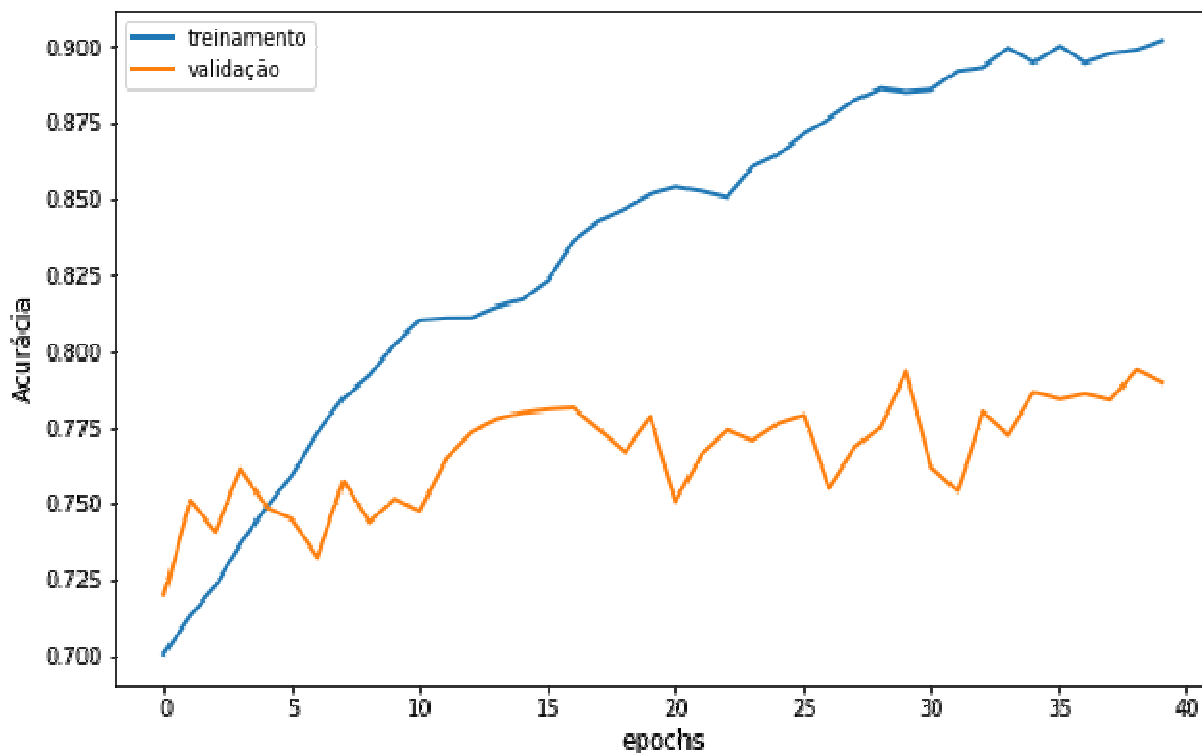


Fonte: Próprio autor

Como foi explicado as imagens foram tratadas de duas maneiras, quanto ao redimensionamento das mesmas, para que todas tivessem o mesmo tamanho, o que é uma restrição da rede. A gráfico da Figura 16 mostra os resultados da acurácia para as imagens que foram mantidas as dimensões originais, e o espaço

restante para chegar aos 256x256 *pixels*, foram preenchidos com a cor preta (ausência de cor), ou seja, o grupo de amostras 1.

Figura 17 Gráfico das acurácias para os dados de treinamento e de validação



Fonte: Próprio autor

A acurácia e perda mostradas no gráfico acima são relativas à execução de uma das iterações do método de validação cruzada, como é possível notar, a acurácia nos dados de treino continua crescente após 40 *epochs*, mas se aproximando de uma estabilização, a curva de crescimento está bem mais plana que no início. Já nos dados de validação o crescimento é bem mais sutil, e oscila bastante, talvez isso se dê por causa da quantidade de imagens serem poucas. A melhor acurácia obtida nos dados de validação foi de 81%.

O processo de validação cruzada foi utilizado para medir a acurácia geral da rede. A tabela 2 mostra os resultados obtidos na validação cruzada para o grupo de amostras 1.

Tabela 2 Acurácia nos dados de validação na validação cruzada

Iteração	Acurácia %
1	78.95
2	76.32
3	78.95
4	78.53
5	78.53
6	76.96
7	76.70
8	78.95
9	78.53
10	78.53
Média: 78.10% Desvio padrão: 1.02	

Fonte: Próprio autor

Embora a acurácia final de cada iteração na validação cruzada tenha chegado no máximo a 78.95%, em alguns *epochs* ela chegou a 81%, como foi possível notar no Gráfico 14.

No conjunto de imagens onde a amostra foi expandida ao tamanho de 256x256 *pixels* os resultados foram abaixo dos resultados do primeiro grupo de amostras. Isso mostra que o melhor método é manter o tamanho original da imagem de preencher o espaço restante com a cor preta (sem cor). Enquanto que no primeiro grupo conseguiu-se uma acurácia de 81%, no grupo 2 a acurácia ficou abaixo dos 70%.

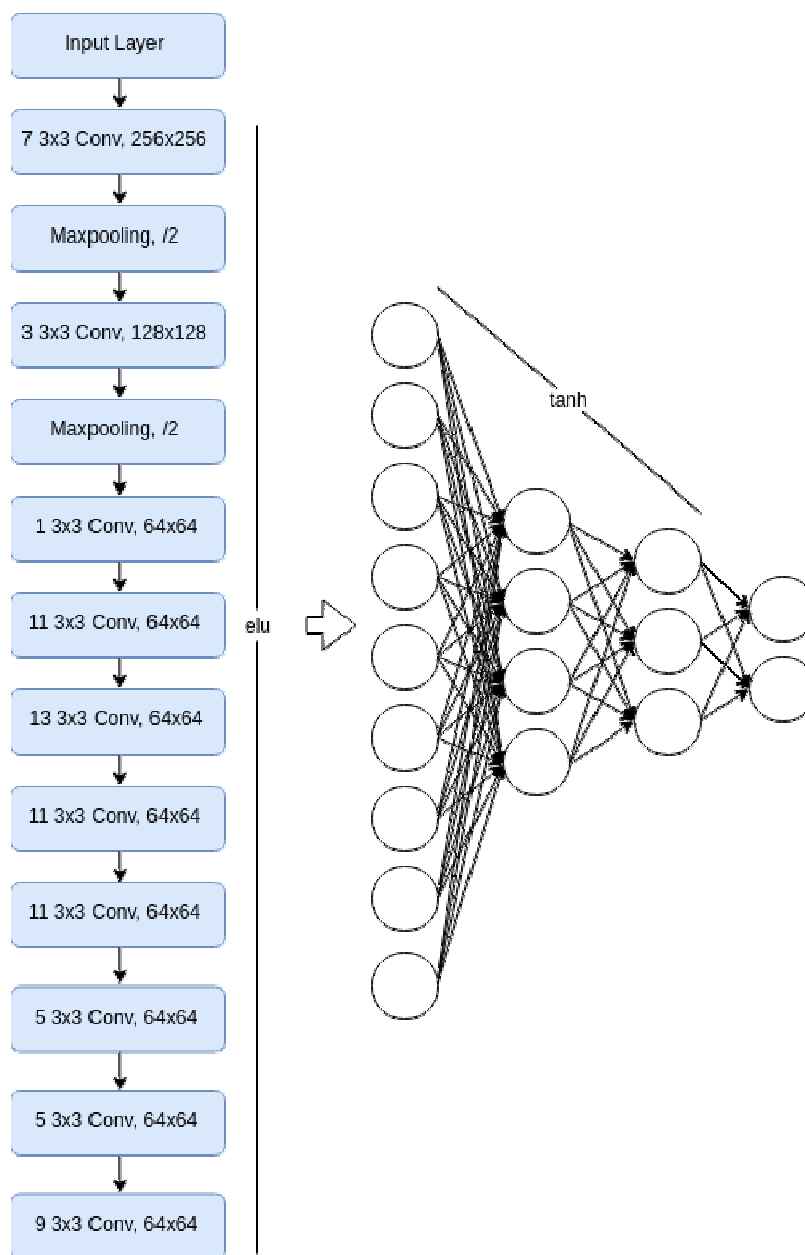
4.1 Modelo desenvolvido

Na Figura 18 é mostrado um diagrama representado o modelo desenvolvido durante esta pesquisa. Os filtros das camadas de convolução foram todos um *kernel*

de 3x3. Nessa parte os parâmetros que o AG passou pelo processo de evolução, foram a quantidade de filtro em cada *layer*. A quantidade de filtros de cada camada convolutiva são: [7, 3, 1, 11, 13, 11, 11, 5, 5, 9]. A quantidade de camadas de *maxpooling* foram duas, a primeira posicionada entre a primeira e segunda camada de convolução, e a segunda posicionada entre a segunda e terceira camada convolutiva. A função de ativação usada foi a *elu*.

No conjunto de camadas densas a configuração ficou da seguinte forma:quatro camadas densas, a primeira com 9 neurônios, a segunda com 4 neurônios, a terceira com 4 neurônios e a quarta (camada de saída) com 2 neurônios. A função de ativação das camadas foi a *tanh* (tangente hiperbólica) e na camada de saída a *softmax*, o otimizador utilizado foi o *binary crossentropy*.

Figura 18 Diagrama do modelo de CNN proposto



Fonte: Próprio autor

O modelo é consideravelmente simples, no entanto, apresenta bons resultados na classificação de cáries dentárias.

5 CONSIDERAÇÕES FINAIS

Sistemas de análise computadorizada de radiografias digitais no uso clínico auxiliam aos profissionais da área odontológica no diagnóstico de doenças nos dentes, mais especificamente no diagnóstico de cáries. Além de ajudarem na redução da demanda de tempo necessário às análises, o que conseqüentemente provoca uma redução de custos.

Nesse trabalho foi apresentado o desenvolvimento de uma arquitetura de rede neural convolucional, com a utilização de algoritmos genéticos como otimizadores dos hiperparâmetros da CNN, para classificação de lesões de cáries em regiões de radiografias digitais. Essa arquitetura pode ser utilizada integrada a um sistema CAD, para auxílio ao diagnóstico.

Foi utilizado um *dataset* com 102 imagens radiográficas, destas foram obtidas 697 amostras de regiões com e sem cáries, as quais foram utilizadas no processo de treinamento e validação da arquitetura proposta. Os resultados obtidos tiveram uma média de acurácia de 78,10% na validação cruzada.

Os objetivos apresentados no início desse trabalho foram alcançados, uma vez que o modelo proposto conseguiu bons resultados na classificação de cáries dentárias. O que implica que pode ser utilizado na construção de uma ferramenta CAD, e dessa forma auxiliar aos profissionais da saúde bucal, na identificação do problema da cárie.

O modelo também pode ser utilizado para classificação em outros problemas envolvendo a análise de radiografias, os quais tenham certa similaridade com as cáries dentárias, ou seja, manchas radiolúcidas.

Para trabalhos futuros pretende-se adquirir uma quantidade maior de amostras, uma vez que se faz necessária uma grande quantidade de imagens, para que a capacidade de generalização de CNNs seja alta. Além disso, pretende-se inserir métodos de detecção automática, para que a detecção das cáries ocorra de forma automática, sem que seja necessário que o odontólogo informe o local desejado à análise.

REFERÊNCIAS

- ALEKSANDER, I; MORTON, H. *An introduction to neural computing*. International Thomson Computer Press, 1995.
- ALMEIDA, L F. Otimização de alternativas para desenvolvimento de campo de petróleo utilizando computação evolucionária. Tese (Engenharia Elétrica) - Pontifícia Universidade Católica Do Rio De Janeiro, 2003.
- BANDURA, O. *Computer vision in dentistry: Using CAD and CBCT systems to detect tooth pathology*. Disponível em:
<https://www.dentistryiq.com/articles/2018/01/computer-vision-in-dentistry-using-cad-and-cbct-systems-to-detect-tooth-pathology.html?cmpid=enl_de_breakthrough_clinical_january-25-2018>.
- FRANCISCO, D O. Métodos baseados em computação evolutiva. 117 p. Dissertação (Engenharia Elétrica) - Escola de Engenharia de São Carlos da Universidade de São Paulo, 2013.
- HAYKIN, Simon; ENGEL, P M. *Redes neurais: princípios e prática*. Artmed, 2007.
- KARPATHY. A. Convolutional Neural Networks (CNNs / ConvNets). Disponível em:
<<http://cs231n.github.io/convolutional-networks/>>. Acesso em: 22/10/2018.
- MARCHIORI, S C. Desenvolvimento de um Sistema para Análise da Estabilidade Transitória de Sistemas de Energia Elétrica via Redes Neurais. 110 p. Tese (Engenharia Elétrica) - Faculdade de Engenharia de Ilha Solteira, Universidade Estadual Paulista, Ilha Solteira, 2006.
- MITCHELL, M. *An introduction to genetic algorithms*. MIT, 2002.
- MØYSTAD, A *et al*. Comparison of standard and task-specific enhancement of Digora® storage phosphor images for approximal caries diagnosis. *Dentomaxillofacial Radiology*, v. 32, n. 6, p. 390–396, 2003.
- PEREIRA, R C R. A digitalização de imagens na radiologia odontológica. 92 p. Monografia (Odontologia) - Universidade Estadual de Campinas. 1999.
- RAWAT, W; WANG, Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation*, v. 29, n. 9, p. 2352–2449, 2017.
- REFAEILZADEH, P; TANG, L; LIU, H. Cross-Validation. *Encyclopedia of Database Systems*, p. 1–7, 2009.
- ROCKENBACH, M I B. Detecção de cáries proximais em radiografias convencionais e digitais: estudo in vitro. Pontifícia Universidade Católica do Rio Grande do Sul, 2006.

ROJAS, R. *Neural networks: a systematic introduction*. Springer, 1996.

RONNEBERGER, O; FISCHER, P; BROX, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Lecture Notes in Computer Science Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, p. 234–241, 2015.

VALIZADEH, S *et al.* Designing of a Computer Software for Detection of Approximal Caries in Posterior Teeth. *Iranian Journal of Radiology*, v. 12, n. 4, Maio de 2015.

VIRDEE, A. Data Augmentation Experimentation. Disponível em: <<https://towardsdatascience.com/data-augmentation-experimentation-3e274504f04b>>. Acesso: 23/12/2018.

WANG, C *et al.* A benchmark for comparison of dental radiography analysis algorithms. *Medical Image Analysis*, v. 31, p. 63–76, 2016.

WATANABE, P. Estado Atual da Arte da Imagem Digital na Odontologia. *Revista da Associação Paulista de Cirurgiões Dentistas*. 53. 320-25, 1999.

WHITE, S C; YOON, D C. Comparative performance of digital and conventional images for detecting proximal surface caries. *Dentomaxillofacial Radiology*, v. 26, n. 1, p. 32–38, 1997.