

Universidade de São Paulo
FFCLRP – Faculdade de Filosofia, Ciências e Letras de Ribeirão
Preto

Samuel Henrique Silva

**Uso de Algoritmos Genéticos e Redes Convolucionais para
classificação de Displasias Corticais Focais em Pacientes com
Epilepsia Refratária**

Ribeirão Preto – SP
2023
Samuel Henrique Silva

Uso de Algoritmos Genéticos e Redes Convolucionais para classificação de Displasias Corticais Focais em Pacientes com Epilepsia Refratária

Versão Original

Dissertação de Mestrado apresentada à Faculdade de Filosofia,
Ciências e Letras de Ribeirão Preto (FFCLRP) da Universidade de
São Paulo (USP), como parte das exigências para obtenção do
título de Mestre em Ciências.

Area de Concentração: Computação Aplicada

Orientador: Renato Tinós

Ribeirão Preto – SP

2023

Use of Genetic Algorithms and Convolutional Networks to Classify Focal Cortical Dysplasias in Patients with Refractory Epilepsy

Original Version

Master's dissertation presented to the Faculty of Philosophy, Sciences and Letters of Ribeirão Preto (FFCLRP) of the University of São Paulo (USP), as part of the requirements for obtaining the title of Master of Science.

Field of Study: Applied Computing

Supervisor: Renato Tinós

Ribeirão Preto – SP

2023

]

Samuel Henrique Silva

Uso de Algoritmos Genéticos e Redes Convolucionais para classificação de Displasias Corticais Focais em Pacientes com Epilepsia Refratária. Ribeirão Preto–SP, 2023.

64p. : il.; 30 cm.

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto da USP, como parte das exigências para a obtenção do título de Mestre em Ciências,

Área: Computação Aplicada.

Orientador: Renato Tinos

1. Redes Convolucionais. 2. Algoritmo Genético. 3. Epilepsia Refratária.

Samuel Henrique Silva

Uso de Algoritmos Genéticos e Redes Convolucionais para classificação de
Displasias Corticais Focais em Pacientes com Epilepsia Refratária

Trabalho aprovado. Ribeirão Preto–SP, XX /XXX/XX:

Orientador:

Orientador

Professor:

Convidado 1

Professor:

Convidado 2

Ribeirão Preto–SP

2023

Agradecimentos

Agradeço este trabalho a todos que sempre me apoiaram nesta jornada do Mestrado, sempre estiveram ao meu lado me apoiando e incentivando a continuar.

Agradeço a minha família, Mae, tios, tias, primas, primos, que sempre esteve muito presente no processo, sempre muito interessados na jornada. Agradeço a minha namorada, Lara Correa, que esteve comigo nessa jornada também sempre me incentivando. Também agradeço meus amigos próximos que torceram muito pela conclusão do mestrado.

Por fim, em especial, agradeço ao meu orientador Renato Tinos, por toda paciência durante o projeto, por todos os valiosos ensinamentos passados, por toda sabedoria para ajudar a conduzir este projeto.

Nós sempre nos definimos pela capacidade de superar o impossível.

Interestelar.

Uso de Algoritmos Genéticos e Redes Convolucionais para classificação de Displasias Corticais Focais em Pacientes com Epilepsia Refratária

Resumo

Displasia Cortical Focal (DCF) é um tipo de lesão cerebral que é a principal causa de Epilepsia Refratária em crianças, e o terceira maior causa em adultos, estando presente em mais de 50% dos casos infantis e 20% nos casos adultos. Pacientes com essa doença sofrem de crises epiléticas os quais os remédios não são capazes de ajudar. O tratamento clínico mais indicado nesse caso é a cirurgia. Mas para ser realizada é necessário se ter uma identificação precisa da lesão. Esse processo é bastante complexo pois a região da lesão não é bem definida para visualização em Imagens de Ressonância Magnética (IRM). Porém com os avanços computacionais sugeriram várias técnicas para o auxílio na classificação de imagens médicas. Neste trabalho é estudado o uso de Algoritmos Genético (AG) com Redes Neurais Convolucionais (CNN) para classificação de imagens com a presença da DCF. O sistema baseado em CNN deve detectar e identificar a localização de DCF nas imagens. Para isso, propõe-se que a CNN classifique janelas retangulares das imagens com o objetivo de localizar regiões afetadas pela DCF. O tamanho das janelas e a sobreposição entre elas tem impacto direto na eficiência do sistema baseado em CNNs. Além disso, os hiperparâmetros influenciam fortemente a performance da CNN. Propõe-se aqui utilizar um algoritmo genético para: i) definir o tamanho das janelas; ii) definir a sobreposição das janelas; iii) definir alguns dos hiperparâmetros da CNN. Como resultado, o AG se mostrou bastante benéfico, otimizando a CNN de maneira a obter uma acurácia superior a 90% na classificação.

Palavras-Chaves: Redes Neurais Convolucionais (CNN), Displasia Cortical Focal (DCF), Imagens de Ressonância Magnética (IRM), Diagnóstico Auxiliado por computador (CAD), Algoritmos Genéticos (AG).

Use of Genetic Algorithms and Convolutional Networks to Classify Focal Cortical Dysplasias in Patients with Refractory Epilepsy

Abstract

Focal Cortical Dysplasia (FCD) is a type of brain injury that is the main cause of Refractory Epilepsy in children, and the third largest cause in adults, present in more than 50% of childhood cases and 20% in adult cases. Patients with this inflammatory disease of epileptic seizures where the drugs are not able to help. The most appropriate clinical treatment in this case is surgery. But to be performed it is necessary if you have an accurate identification of the injury. This process is quite complex because the region of the lesion is not well defined for viewing on Magnetic Resonance Images (MRI). However, with the computational advances, have been suggested several techniques to aid in the classification of medical images. In this work, the use of Genetic Algorithm with Convolutional Neural Networks (CNN) to classify images with the presence of FCD is studied. The CNN-based system should detect and identify the location of FCD in the images. For this, it is proposed that CNN classify rectangular windows of the images in order to locate regions affected by FCD. The size of the windows and the overlap between them has a direct impact on the efficiency of the CNN-based system. In addition, hyper-parameters strongly influence CNN's performance. It is proposed here, using a genetic algorithm to: i) define the size of the windows; ii) define the overlap of the windows; iii) define some of CNN's hyper-parameters. As a result, the AG proved to be quite beneficial, optimizing the CNN in order to obtain an accuracy greater than 90% in the classification

Keywords: Convolutional Neural Networks (CNN), Focal Cortical Dysplasia (FCD), Magnetic Resonance Images (MRI), Genetic Algorithm (GA).

LISTA DE TABELAS

| | |
|---|----|
| TABELA 1: MATRIZ DE CONFUSÃO OBTIDA PELA CNN..... | 31 |
| TABELA 2: MATRIZ DE CONFUSÃO OBTIDA PELO MLP | 32 |
| TABELA 3: EVOLUÇÃO DO AG PARA OTIMIZAÇÃO DOS QUADRANTES | 34 |
| TABELA 4: RESULTADO AG NA PRIMEIRA EXECUÇÃO..... | 36 |
| TABELA 5: RESULTADO AG NA SEGUNDA EXECUÇÃO | 36 |
| TABELA 6: RESULTADO AG NA TERCEIRA EXECUÇÃO | 36 |
| TABELA 7: RESULTADO AG NA QUARTA EXECUÇÃO..... | 37 |
| TABELA 8: RESULTADO AG NA QUINTA EXECUÇÃO..... | 37 |

Lista de Imagem

| | |
|---|----|
| FIGURA 1: PROCESSO ILUSTRADO DAS OPERAÇÕES REALIZADAS NA CAMADA DE CONVOLUÇÃO:..... | 7 |
| FIGURA 2: RESULTADO DA APLICAÇÃO DO MAX POOLING EM UMA MATRIZ 4x4. | 8 |
| FIGURA 3: FIGURA ILUSTRA O PROCESSO DE OTIMIZAÇÃO REALIZADO POR ALGORITMO GENÉTICO UTILIZADA NESTE PROJETO. FONTE: AUTOR..... | 11 |
| FIGURA 4 : IMAGEM A: IMAGEM NIFIT SEM A PRESENÇA DO LABEL. IMAGEM B: IMAGEM NIFIT COM A PRESENÇA DO LABEL..... | 13 |
| FIGURA 5. IMAGEM A: ENTRADA DO PROCESSAMENTO. IMAGEM B: SAÍDA DO PROCESSAMENTO..... | 15 |
| FIGURA 6: ILUSTRAÇÃO DO PROCESSO DE STRIDE. FONTE: AUTOR..... | 16 |
| FIGURA 7: JANELA CRIADA COM DCF PRESENTE SINALIZADA PELO LABEL. FONTE: AUTOR | 19 |
| FIGURA 8: ILUSTRAÇÃO DO VETOR DE CARACTERÍSTICA DE UM AG. FONTE: AUTOR.. | 21 |
| FIGURA 9: FIGURA ILUSTRANDANDO PROCESSO DE AVALIAÇÃO DO ALGORITMO GENÉTICO. FONTE: AUTOR..... | 22 |
| FIGURA 10: ILUSTRAÇÃO REPRESENTA UMA REDE CONVOLUCIONAL COM 3 CAMADAS COM 64,128, 256 NEURÔNIOS POR CAMADA RESPECTIVAMENTE. TODAS COM RELU COMO FUNÇÃO DE ATIVAÇÃO, KERNEL E MAXPOOL 2x2 E DROPOUT DE 0.2. FONTE: AUTOR | 25 |
| FIGURA 11: DISPOSIÇÃO DAS POSSÍVEIS CLASSIFICAÇÕES DE DADOS UTILIZANDO A MATRIZ DE CONFUSÃO..... | 27 |
| FIGURA 12: GRAFICO DE CURVA ROC. | 28 |
| FIGURA 13: CURVA ROC PARA CNN FONTE: AUTOR..... | 31 |

| | |
|--|----|
| FIGURA 14: REPRESENTAÇÃO DA CURVA ROC. FONTE: AUTOR..... | 32 |
| FIGURA 15: A FIGURA ILUTRA A EVOLUÇÃO DOS VALORES DE LARGURA E ALTURA OTIMIZADOS PELO AG. FONTE: AUTOR..... | 35 |
| FIGURA 16: FIGURA ILUSTRA A EVOLUÇÃO DO FITNESS AO LONGO DAS GERAÇÕES. FONTE: AUTOR | 35 |
| FIGURA 17: ILUSTRAÇÃO DA EVOLUÇÃO DOS VALORES MÉDIOS DE LARGURA E ALTURA DAS 5 EXECUÇÕES. FONTE: AUTOR..... | 38 |
| FIGURA 18: A FIGURA ILUSTRA EVOLUÇÃO DO NÚMERO DE NEURONIOS DA CAMADA OCULTA JUNTAMENTE COM O NÚMERO DE NEURONIOS DA CAMADA DENSA DOS VALORES MÉDIOS DAS 5 EXECUÇÕES. FONTE: AUTOR | 38 |
| FIGURA 19: A FIGURA 19 ILUSTRA A EVOLUÇÃO DE FITNESS E ACURÁCIA DOS VALORES MÉDIO DAS CINCO EXECUÇÕES. FONTE: AUTOR | 39 |
| FIGURA 20: ARQUITETURA DA CNN OTIMIZADA PELO AG. | 40 |
| FIGURA 21: ARQUITETURA DA CNN ESTÁTICA. | 40 |
| FIGURA 22: COMPARATIVO DOS VALORES DE F1-SCORE E ACURÁCIA ENTRE O MLP, A CNN ESTÁTICA, O MELHOR INDIVIDUO DAS CINCO EXECUÇÕES DA CNN COM O AG E O RESULTADO DA MÉDIA DAS 5 EXECUÇÕES. FONTE: AUTOR | 41 |

Lista de Siglas e Abreviações

| | |
|------|--|
| DAE | Drogas Antiepiléticas |
| DCF | Displasia Cortical Focal |
| ILAE | <i>International League Against Epilepsy</i> |
| IRM | Imagem de Ressonância Magnética |
| RNA | Redes Neurais Artificiais |
| CAD | <i>Computer-Aided Diagnosis</i> |
| CNN | <i>Convolutional Neural network</i> |
| AG | Algoritmo Genético |
| RM | Ressonância Magnética |

Sumário

| | | |
|-------|---|----|
| 1 | Introdução | 1 |
| 1.1 | Revisão Bibliográfica..... | 3 |
| 1.2 | Objetivos | 5 |
| 2 | Referenciais Teóricos..... | 6 |
| 2.1 | Redes Neurais Convolucional (CNN)..... | 6 |
| 2.1.1 | Camadas de Convolução..... | 6 |
| 2.1.2 | Mapa de características..... | 7 |
| 2.1.3 | Camadas de Pooling..... | 7 |
| 2.1.4 | Camadas totalmente conectada | 9 |
| 2.2 | Algoritmo Genético..... | 9 |
| 3 | Métodos..... | 12 |
| 3.1 | Base de dados | 12 |
| 3.2 | Processamento da Base de Dados..... | 14 |
| 3.2.1 | Obtendo Slices..... | 14 |
| 3.2.2 | Técnica do Janelamento..... | 14 |
| 3.2.3 | Corte e marcação das janelas | 15 |
| 3.2.4 | Seleção de Região..... | 16 |
| 3.2.5 | Aumento da Base de Dados | 17 |
| 3.2.6 | Treino, teste e validação..... | 18 |
| 3.3 | Algoritmo Genético..... | 19 |
| 3.3.2 | Algoritmos genéticos e as janelas..... | 19 |
| 3.3.3 | <i>Stride</i> | 19 |
| 3.3.4 | Algoritmos genéticos e os hiper-parâmetros da CNN | 20 |
| 3.3.5 | Indivíduo | 20 |
| 3.3.6 | Função de fitness..... | 21 |
| 3.3.7 | Elitismo e Torneio | 23 |
| 3.3.8 | Mutação e Crossover | 23 |
| 3.4 | Rede Convolucional | 24 |

| | |
|--|----|
| 3.4.1 A estrutura da camada convolucional | 24 |
| 3.4.2 A relação da camada oculta com a camada densa | 25 |
| 3.4.3 Estrutura estática da rede | 26 |
| 3.5 Análise dos resultados | 26 |
| 3.6 Modelos Utilizados Para Comparação | 29 |
| 4 Resultados | 30 |
| 4.1 Comparação de modelos utilizando janelas fixas e sem sobreposição. . | 30 |
| 4.1.1 Resultado da CNN Estática | 30 |
| 4.1.2 Resultado MLP | 31 |
| 4.2 Implementação do AG para otimização do tamanho das janelas | 32 |
| 4.3 Resultado do Modelo Proposto (CNN e janelas otimizadas pelo AG)..... | 35 |
| 4.3 Discussão | 42 |
| 4.5 Limitações | 43 |
| 5 Conclusão | 45 |
| 6 Referencias | 47 |

1

Introdução

A Epilepsia é uma doença crônica que afeta aproximadamente 50 milhões de pessoas no mundo (GARZON, 2002). O tratamento da doença é realizado com o uso de Drogas Antiepilépticas (DAEs), sendo que em 70% a 80% dos casos o tratamento com remédios consegue controlar as crises epiléticas sofridas pelo paciente (WAN, 2000). Apesar da alta eficiência das DAEs existem casos no qual não é possível controlar a crise. Nesse cenário, a doença é chamada de epilepsia refratária. Pacientes nessas condições sofrem de convulsões longas e inesperadas, podendo levá-los a hospitalização e aumento do risco de morte súbita (FRENCH, 2007). Assim, nessas circunstâncias o tratamento cirúrgico mostra-se mais efetivo para o controle da doença. Porém a cirurgia só pode ser realizada quando se tem a localização precisa da região cerebral lesionada, isto é, a área exata responsável pela crise. O diagnóstico desta região é muitas vezes difícil de ser feito, principalmente em casos de lesões sutis como a Displasia Cortical Focal (DCF).

A DCF é a malformação do desenvolvimento cortical (CAMARGO et al., 2010). Graças aos avanços tecnológicos sabemos que as DCFs são a principal causa da Epilepsia Refratária em crianças, estando presente em quase 50% das crianças submetidas a cirurgia em Epilepsia, e sendo a terceira causa mais frequente em adultos, chegando a 20% dos casos (KRSEK *et al.*, 2008). Além disso, existem diferentes tipos de DCFs, que por sua vez resultam em diferentes características clínicas e em imagens médicas (KRSEK *et al.*, 2008).

A atual classificação utilizada foi proposta pela *International League Against Epilepsy* (ILAE). Segundo a ILAE existem 3 tipos distintos de DCF, classificadas como tipo I, II ou III. Neste trabalho, trataremos do problema de classificação de DCFs do tipo II, que tem como característica principal a

presença de células anormais, chamadas de neurônios displásicos ou células balão (PALMINI *et al.*, 2004).

O diagnóstico de lesões DCF é feito principalmente utilizando-se imagens de Ressonância Magnética (IRM). Apesar dos avanços alcançados nos últimos anos neste tipo de imagem, a classificação de DCFs do tipo II é complexa, visto que apresentam bordas irregulares e tecido saudável de má resolução, fazendo com que o diagnóstico se torne muito complexo dada a confusão entre a lesão e o tecido saudável. Portanto, mesmo quando a lesão é identificada, encontrar seus limites na IRM torna-se uma tarefa de difícil execução. Deste modo, técnicas de auxílio ao diagnóstico por imagens podem ajudar na classificação das lesões.

Graças principalmente aos poderosos avanços obtidos nos processadores das máquinas atuais, o uso de Redes Neurais Artificiais (RNAs) se torna cada vez mais comum para solução de problemas de classificação. As RNAs são importantes modelos computacionais inspirados no funcionamento básico de conjuntos de neurônios no cérebro humano. Diversos tipos de RNA foram desenvolvidos, tendo hoje as mais diversas aplicações. O uso de RNAs na área de imagens médicas vem se mostrando útil no auxílio ao diagnóstico e tomada de decisão.

O Diagnóstico Auxiliado por Computador (“Computer-Aided Diagnosis” - CAD) tem como objetivo melhorar a precisão dos diagnósticos realizados, uma vez que o computador pode captar padrões que para os humanos são difíceis de serem detectados. O desenvolvimento de sistemas CAD não tem como objetivo substituir o médico em momento algum, mas apenas auxiliá-lo na tomada de decisão. O uso desses sistemas na radiologia tem se mostrado bastante promissor, principalmente para detecção e classificação de lesões, uma vez que eles aumentam o desempenho e a produtividade do radiologista. (AZEVEDO-MARQUES, 2001).

Tradicionalmente em sistemas CAD, imagens sofrem diversos tipos de pré-processamento, como por exemplo a segmentação, e extração de características (atributos) antes da aplicação dos classificadores. Este trabalho tem como objetivo utilizar modelos de Deep Learning, mais especificamente uma Rede Neural Convolucional, para detecção de Displasia Cortical Focal em

Imagens de Ressonância Magnética sem realizar nenhum pré-processamento nestas imagens e sem a extração de características de forma manual.

1.1 Revisão Bibliográfica

SIMOZO (2018) em seu trabalho buscou realizar a classificação de DCF através da análise de espessura cortical e padrões de textura em imagens estruturais de RM. A base utilizada contava com imagens de DCF confirmada por meio de uma análise histológica e uma segmentação manual feita por um especialista. A técnica utilizada tinha foco na utilização de métodos de pré-processamento e extratores de características, para então classificar as imagens. Os voxels da figura eram isoladamente classificados por técnicas de aprendizado de máquina utilizando características (filtros) pré-definidos. Seu estudo obteve êxito conseguindo classificar com precisão 81% das imagens.

WANG et al (2020) utilizam, assim como neste trabalho, CNNs para classificar DCF em IRM. Para isso, os autores aplicam pré-processamento nas imagens antes da utilização da rede. No pré-processamento realizado é corrigido o campo de polarização da imagem, e aplica-se a normalização das intensidades das cores e a remoção de tecidos não cerebrais. As imagens são divididas em retângulos e então levadas a rede, sendo ela composta por 5 camadas convolucionais e 2 camadas densas. Como resultado, obtém-se uma sensibilidade superior a 95%.

DEV et al (2019) propuseram em seu estudo realizar a classificação de DCF utilizando uma arquitetura U-Net. A Base de dados utilizada foi composta com IRM de 40 pacientes. Para melhorar a qualidade das imagens, dois tipos de pré-processamento foram aplicados. Como IRM normalmente são afetadas por ruídos, primeiramente eles submetem as imagens a um processo de remoção de ruído. Logo após é removido todo conteúdo das imagens que não sejam relacionadas ao cérebro, sobrando assim apenas partes relevantes ao estudo. Em seguida, a U-Net foi treinada, obtendo-se a classificação correta de 33 dos 40 pacientes.

SILVA (2019) em seu trabalho de conclusão de curso utilizou a mesma base de (SIMOZO, 2018) para estudar a classificação de DCFs em imagens de RM através do uso de CNN. Diferentemente do trabalho apresentado em (SIMOZO, 2018), no estudo não foi aplicado nenhum pré-processamento nas imagens. As imagens foram divididas em quadrados (janelas), sem sobreposição e com tamanho fixo definido a priori. Então, as janelas foram individualmente classificadas pela CNN. A CNN apresentou bons resultados, sendo que o trabalho (SILVA, 2019) serve de base para o desenvolvimento do método proposto neste mestrado.

DEL LAMA et al (2022) em seu projeto de mestrado desenvolveu um sistema CAD para classificação de Fraturas Vertebrais por Compreensão (FVC). A classificação foi feita utilizando um CNN e um Algoritmo Genético (AG) para selecionar os atributos e os hiperparâmetros da CNN. Através da seleção de atributos realizada pelo AG, foi possível analisar aqueles mais relevantes para a classificação das imagens. A CNN otimizada pelo AG no estudo se mostrou estatisticamente superior às demais CNNs utilizadas no estudo, demonstrando a eficiência do uso de uma AG para seleção de bons parâmetros de forma automática.

RODRIGUES (2018) propôs em seu estudo o uso de AG para redução de dimensionalidade em um sistema CAD para diagnóstico de Câncer de Mama em imagens termográficas. Seu estudo ressalta a importância de bons parâmetros do AG, principalmente a taxa de mutação. A mutação oferece ao algoritmo um comportamento exploratório, impedindo que os indivíduos fiquem presos em máximos locais, através da variabilidade e renovação que é aplicada por ele. Foram obtidos bons resultados na pesquisa, reduzindo consideravelmente o número de atributos utilizados pelo classificador.

SILVA (2017) propôs em seu estudo o uso de AG para otimização de filtros das camadas convolucionais e do número de neurônios da camada densa de um CNN. Buscava-se otimizar a rede para melhorar a classificação de nódulos pulmonares em tomografias computadorizadas, classificando-os como benignos ou malignos. Com o uso do AG foi possível encontrar melhores CNNs e evitar a estagnação dos modelos. Como função de avaliação (fitness) do AG, utilizou-se uma soma de métricas retiradas da matriz de confusão (sensibilidade,

especificidade, acurácia). A fim de buscar um modelo com maior assertividade para os nódulos malignos, atribuiu-se peso maior para sensibilidade. Ao fim do estudo foi obtido um resultado superior a 90% em todas as métricas avaliadas.

1.2 Objetivos

As RNAs vêm cada vez mais ganhando espaço na área da saúde. Anteriormente (SILVA, 2019), propusemos o uso de CNNs para detectar e identificar a localização de DCF em imagens de ressonância magnética. Para isso, as imagens eram divididas em janelas retangulares que eram então classificadas pela CNN. A divisão das imagens em janelas tinha o objetivo de permitir a localização das regiões afetadas pela DCF. As janelas tinham tamanhos pré-definidos e não possuíam sobreposição. Os hiperparâmetros, que influenciam fortemente a performance da CNN, eram também pré-definidos.

O objetivo principal deste trabalho é aprimorar o sistema de detecção e localização das DCFs nas imagens por meio de: i) uso de janelas com sobreposição; ii) uso de um Algoritmo Genético (AG) para otimizar concomitantemente a sobreposição e o tamanho das janelas, e os hiperparâmetros da CNN. Entre os hiperparâmetros da CNN, em destaque especial é dado para aqueles que controlam a arquitetura da rede neural artificial.

O trabalho anterior (SILVA,2019) mostrou que CNNs podem ser utilizadas para classificar janelas obtidas em imagens de ressonância magnética para a identificação de DCFs. Neste contexto, a hipótese investigada aqui é que o uso de algoritmos para otimizar as janelas das imagens e os hiperparâmetros da CNN deve gerar classificadores mais eficientes do que os métodos atuais nos quais CNN ou classificadores tradicionais são utilizados.

Assim, o trabalho tem entre os objetivos secundários:

- Propor e testar o uso do sistema com sobreposição de janelas;
- Propor e testar classificadores tradicionais baseados em extração de características (filtros) para o problema investigado;
- Analisar e comparar os sistemas propostos;
- Investigar as características selecionadas pelo AG

2

Referenciais Teóricos

2.1 Redes Neurais Convolutacional (CNN)

2.1.1 Camadas de Convolução

A camada de convolução pode ser a primeira camada após a entrada na rede, sendo que nela é realizado o maior número de operações presentes na CNN, que são realizadas pelos filtros (também chamados de *Kernels*). Podemos dizer que os filtros presentes nessa camada têm função análoga aos neurônios presentes nas RNAs tradicionais. Uma CNN pode uma ou mais camadas de convolução presentes na sua arquitetura.

Ao aplicar um filtro na imagem, obtém-se o mapa de características que representa a imagem. Nesta etapa diferentes filtros são aplicados à mesma imagem obtendo assim diferentes mapas de características. Desta forma a rede consegue aprender informações importantes presentes nas imagens.

Um filtro convolutacional basicamente é uma matriz, onde cada elemento representa um peso. O filtro percorre a imagem, obtendo-se sucessivamente a soma das multiplicações entre os pesos e as respectivas entradas, como descrito na Figura 1, gerando assim a operação de convolução. Ao aplicar esta operação em todos os pixels da imagem é obtido uma nova matriz, conhecida como mapa de ativação, resultando assim no mapa de características associado ao filtro (neurônio). Os valores deste mapa são submetidos a uma função de ativação. Este passo tem a finalidade de compor o valor da saída de maneira que a eficiência das CNNs melhore para problemas não-lineares diversos (LIMA, 2019). Atualmente é preferível que se use a ReLU (Unidade de retificação linear) como função de ativação (CLEVERT et al, 2015) por ela ser facilmente computada.

Na camada convolucional também definimos o número de filtros que a imagem é submetida, podendo variar conforme a necessidade. E também qual o tamanho do filtro do qual a imagem é submetida, sendo a única necessidade desse filtro é que seja uma matriz quadrada e tenha o número de dimensões igual à quantidade de canais da imagem. Figura 1 ilustra o processo da camada de convolução.

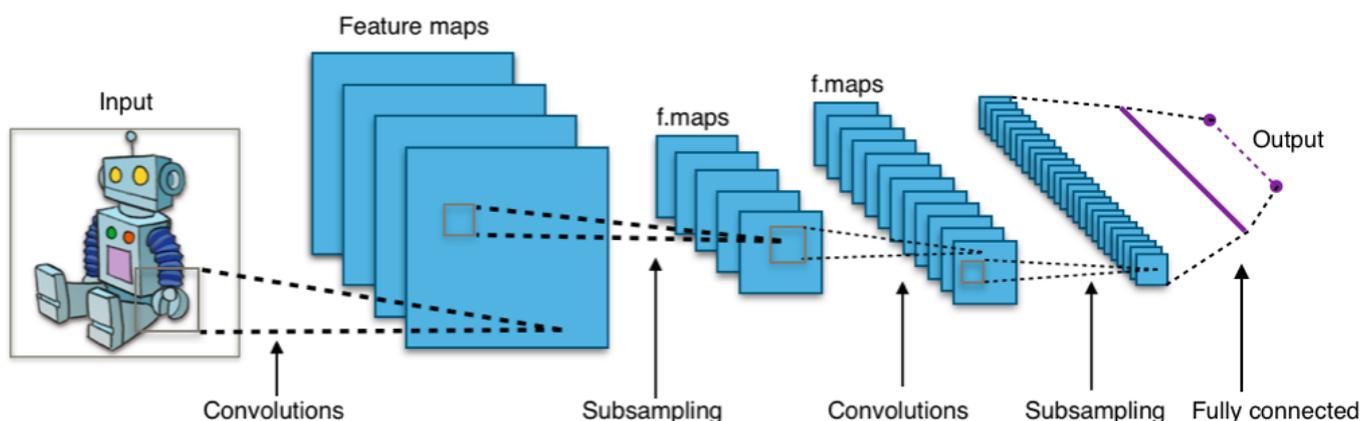


Figura 1: Processo ilustrado das operações realizadas na camada de convolução:

Fonte: Aphex34, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=45679374>

2.1.2 Mapa de características

Chamamos de mapa de características o resultado obtido após a imagem ter passado por uma camada de convolução. O resultado obtido é uma representação da imagem onde as principais características da imagem que foram ressaltadas pelos filtros estão presentes. Portanto, na camada anterior, ao aplicar diferentes filtros em uma imagem, é obtido as diferentes características presentes na imagem, e essas características serão usadas como entrada da próxima camada.

2.1.3 Camadas de Pooling

Esta camada tem como objetivo reduzir a quantidade de dados resultante da camada de convolução. Assim, uma camada de pooling é geralmente aplicada após cada camada de convolução. Reduzir a dimensionalidade, neste caso, é

geralmente benéfico pois ajuda a diminuir o número de parâmetros ajustáveis, dados processados pela rede e controlar o *overfitting* que pode ocorrer durante o treinamento do modelo.

Uma das técnicas de *pooling* mais utilizadas é a *maxpooling*, que apresenta uma grande capacidade de extração de características. Nesta técnica o filtro que percorrerá o mapa de característica selecionará o maior valor da janela, para assim compor uma nova matriz de saída. Esta situação está ilustrada na Figura 2, onde temos como exemplo um mapa de características de tamanho 4x4, um filtro de *maxpooling* de 2x2 e passo (*stride*) 2, obtendo assim como saída desta camada uma nova matriz no formato 2x2, com os valores mais significativos que estavam presentes na matriz de entrada.

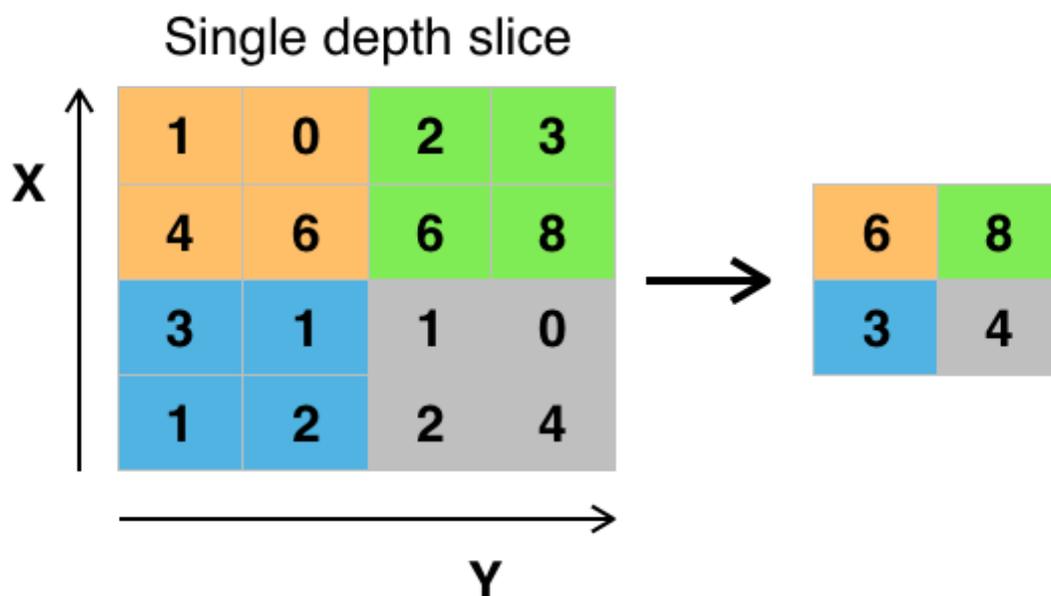


Figura 2: Resultado da aplicação do Max Pooling em uma matriz 4x4.

Fonte:

https://commons.wikimedia.org/wiki/File:Max_pooling.png#/media/File:Max_pooling.png

2.1.4 Camadas totalmente conectada

Por fim, nas últimas camadas são frequentemente utilizadas camadas totalmente conectadas, semelhantes às camadas empregadas em uma rede neural multicamadas do tipo Perceptron Multicamadas (Multilayer Perceptron - MLP). Seu funcionamento dentro das CNNs é idêntico ao seu uso do MLP. Os neurônios desta camada possuem pesos, os quais serão ajustados com o algoritmo supervisionado de *backpropagation* (ROJAS, 1996). No aprendizado supervisionado, o erro (diferença entre valor desejado e saída obtida pelo neurônio na última camada) além de ser propagado internamente aos neurônios das camadas totalmente conectadas é também propagado para ajustar os pesos dos filtros de convolução. Nos neurônios das camadas totalmente conectadas geralmente é utilizada como função de ativação a função *softmax* quando se tem mais de duas classes e a sigmoideal para problemas binários. Neste último caso, o número de neurônios da saída deve ser igual ao número de classes presente no problema.

As camadas totalmente conectadas recebem como entrada um vetor e, como visto anteriormente as camadas de convolução e *pooling* oferecem como saída uma matriz. Logo, é necessário que a matriz produzida pela última camada antes das camadas totalmente conectadas seja transformada em um vetor, sendo esse processo conhecido como *flatten*.

2.2 Algoritmo Genético

Algoritmo Genético (AG) é uma técnica computacional inspirada na Teoria da Evolução por Seleção Natural proposta por Charles Darwin. Sua principal aplicação é em problemas de busca e otimização (LINDEN,2008; EIBEN,2003).

No AG, as soluções candidatas para os problemas de otimização são codificadas em vetores chamados de cromossomos. Um indivíduo do AG armazena um cromossomo junto com a avaliação da solução (fitness). O AG submete uma população inicial composta por um número fixo de indivíduos a operadores de reprodução e seleção inspirados em mecanismos encontrados na genética e na evolução por seleção natural. Esses operadores são aplicados a

todos os indivíduos da população, sendo, e o processo repetido por N gerações (iterações) ou até que outro critério de parada seja alcançado. Os indivíduos obtidos ao final de cada geração são avaliados de acordo com o que se deseja otimizar no problema dado, obtendo-se então o fitness do indivíduo.

Como operador de reprodução uma das técnicas mais utilizadas é a de crossover, que basicamente consiste na troca de genes (elementos do vetor que representa o cromossomo) entre dois indivíduos pais. Os indivíduos pais são selecionados por meio de operadores de seleção. A troca realizada entre os indivíduos no crossover é realizada por meio de regras pré-definidas, e a ocorrência ou não de crossover para cada par de pais é determinada pela taxa de crossover. Outro operador de transformação que é geralmente empregado é a mutação. Na mutação, genes do cromossomo do indivíduo são transformados aleatoriamente. Estas transformações ocorrem com frequência determinada pela taxa de mutação.

A seleção de indivíduos é aplicada com o intuito de selecionar os pais que serão submetidos aos operadores de reprodução. Uma operação de seleção muito utilizado é o método da roleta, no qual os indivíduos com melhores avaliações (*fitness*) têm uma probabilidade maior de compor a população da próxima geração. Outra operação possível para seleção é chamada elitismo, esse método busca garantir que os indivíduos com maior *fitness* estejam na próxima geração. Além dessas, outra técnica bastante utilizada é a do torneio, neste método são sorteados ao acaso dois indivíduos e o mais apto, maior fitness, é levado para próxima geração. A figura 3 ilustra os processos de AG implantados neste projeto.

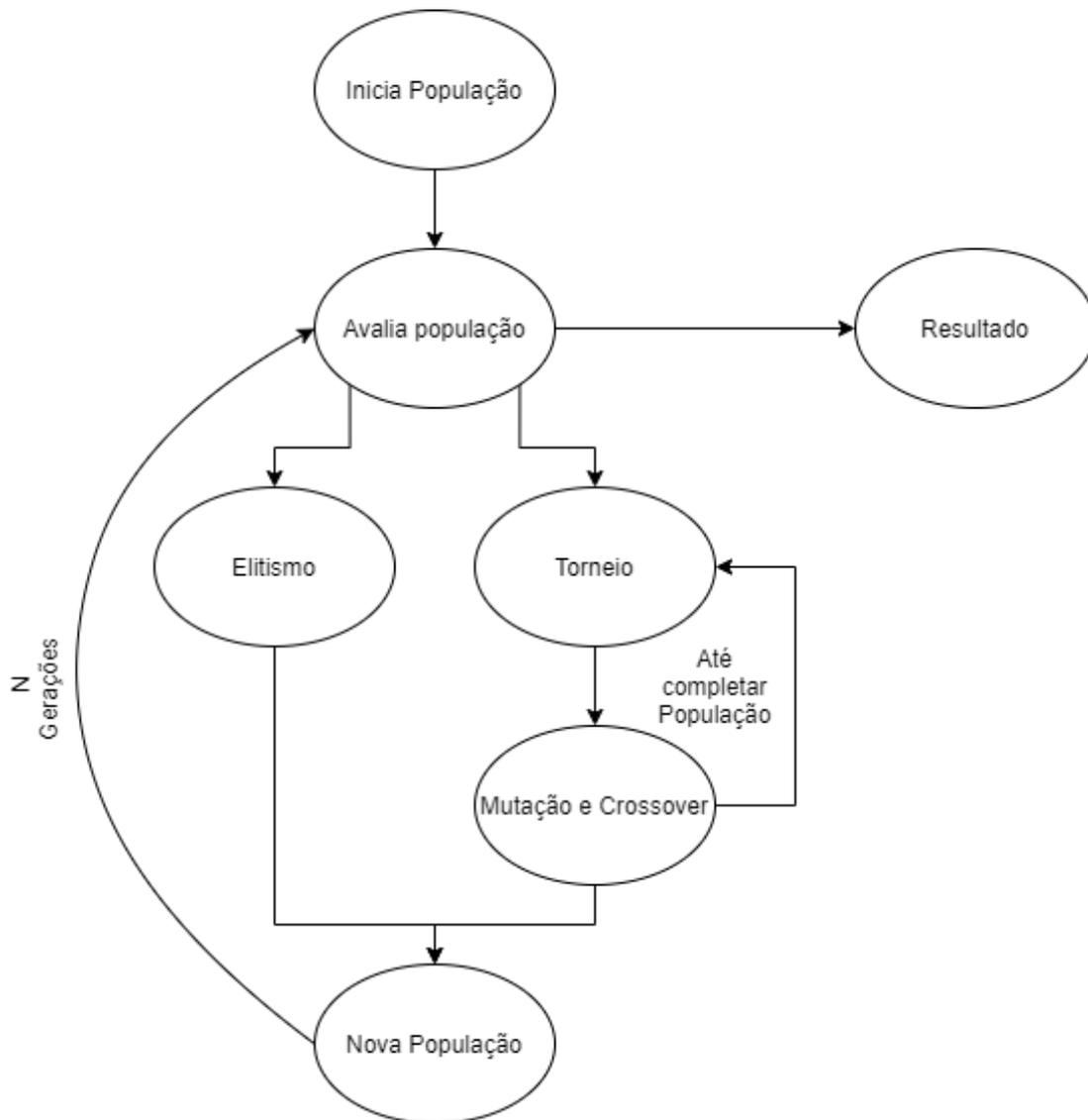


Figura 3: Figura ilustra o processo de otimização realizado por algoritmo Genético utilizada neste projeto. Fonte: Autor

3

Métodos

3.1 Base de dados

A base de dados deste projeto é composta por Imagens de Ressonância Magnética (IRM) do cérebro de sete pacientes, sendo quatro do sexo masculino e três do sexo feminino, todos diagnosticados com DCF.

As imagens foram adquiridas e disponibilizadas pelo Centro de Cirurgia de Epilepsia (CIREP) do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto (HCFMRP) no formato *Neuroimaging Informatics Technology Initiative* (NIFIT), e possuem autorização para uso em pesquisas. Esta utilização foi previamente aprovada pelo Comitê de Ética e Pesquisa da Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, CAAE: 63475616.7.0000.5407, aprovado com Parecer Consubstanciado de número 1.984.671, com o Hospital das Clínicas de Ribeirão Preto como instituição co-participante.

O estudo foi realizado utilizando as imagens no corte axial e ponderadas em T1. Para cada imagem presente foi disponibilizado um *label*, isto é, uma marcação onde a área com DCF foi segmentada. Essa marcação foi realizada por um neurologista experiente utilizando o software 3D *Slicer*. Na Figura 4 ,

vemos em (A) imagem sem o *label* e em (B) a imagem com o *label*.



Figura 4 : Imagem A: imagem NIFIT sem a presença do *label*. Imagem B: Imagem NIFIT com a presença do *label*.

Fonte : Autor

A base de dados já foi utilizada em outros projetos anteriores (SIMOZO,2018; SILVA,2019). Um dos objetivos da tese de Simozo era realizar a classificação das imagens com DCF. Diferentemente deste projeto, a classificação foi realizada através da extração manual de características das imagens, onde foram utilizados filtros para extrair medidas como: Energia, Homogeneidade, Intensidade, Correlação de Haralik, Entropia, correlação, entre outros. Em seu trabalho é possível encontrar todos os filtros utilizados.

Os exames estão anonimizadas, garantindo assim a privacidade e integridade dos pacientes estudados. Os exames também passaram por pré-processamento, sendo este o mesmo foi feito no projeto de conclusão de curso (SILVA, 2019) e apresentado a seguir. Vale ressaltar que pouco pré-processamento é utilizado neste trabalho e ele é feito todo de maneira automática. Por exemplo, não é aplicada segmentação das imagens e nenhuma correção é feita.

3.2 Processamento da Base de Dados

3.2.1 Obtendo Slices

As sete imagens no formato NIFIT são imagens tridimensionais. Sendo assim, para uso nas CNN neste projeto, serão utilizados *slices* (cortes) da imagem de cada paciente, das imagens com e sem *label*, sendo cada corte com 0.3 mm de distância entre cada *slice*. Os cortes são realizados apenas em regiões onde exista presença de DCF; portanto foi gerado uma quantidade diferente de *slices* para cada paciente.

Após a obtenção dos *slices*, a partir das 7 imagens NIFIT foi obtido 41 imagens jpg com *label* e 41 sem *label*, ambas ponderadas em T1. Os cortes realizados nas imagens são feitos utilizando o software 3D *Slice*.

3.2.2 Técnica do Janelamento

Neste trabalho é utilizada a técnica do janelamento. A técnica consiste em marcar e depois cortar a imagem em segmentos menores e utilizar estes segmentos como entradas das redes, sendo estes segmentos chamados de janelas. O uso desta técnica é necessário neste estudo pois assim é possível, além de classificar a DCF, detectar a sua localização na imagem. Pode-se assim indicar ao médico a região onde a CNN indica maior probabilidade de existência de DCF. O corte nas imagens é realizado apenas nas regiões da imagem onde o cérebro está presente, isto é, é utilizado um limiar para remover automaticamente as outras regiões. Na figura abaixo vemos a imagem (A) de

entrada e a imagem (B) de saída após o processo de janelamento com janelas quadradas sem sobreposição (SILVA, 2019).

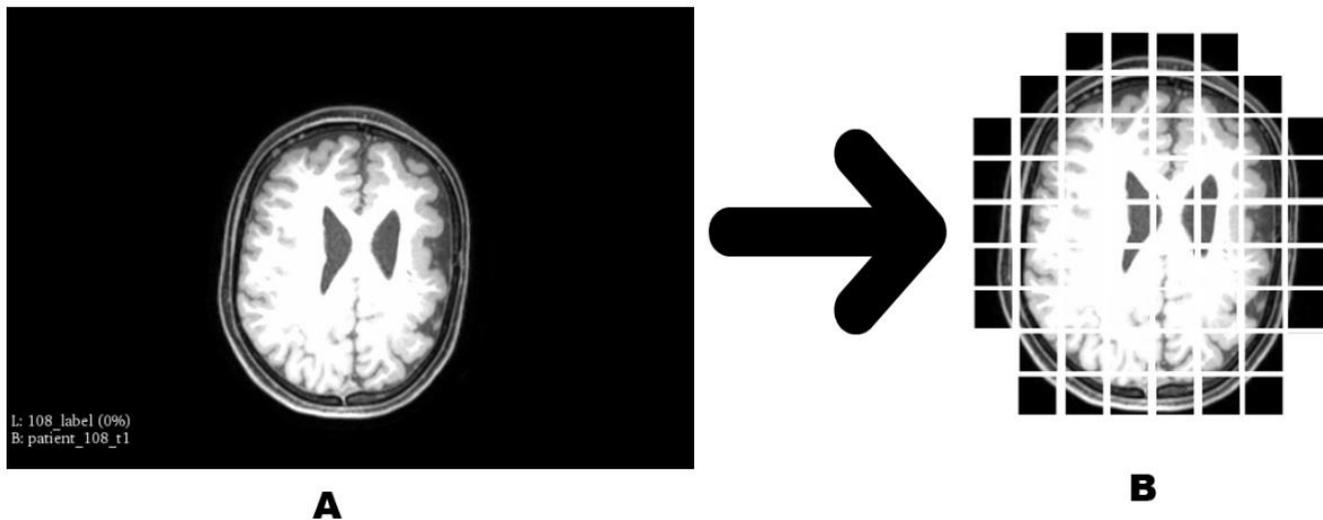


Figura 5. Imagem A: Entrada do processamento. Imagem B: Saída do processamento.

Fonte: Autor

Como mostrado anteriormente existem dois tipos de imagem na base de dados, imagens com *label* e imagem sem *label*. A necessidade desses dois tipos de imagens se dá pois apenas nas imagens com *label* é possível se obter a localização da DCF na imagem. Sendo assim as imagens com *label* foram utilizadas para fazer atribuição de classes de maneira corretas nas bases. O código utilizado para esta atribuição foi desenvolvido em (SILVA,2019), basicamente para cada janela criado nas imagens sem *label* é criado uma janela homologa nas imagens com *label*, para atribuição de classes a cada janela é verificada se sua janela homologa tem *label* ou não. No projeto anterior, é importante ressaltar que o tamanho das janelas era fixo, diferente deste que é dado pelo Algoritmo Genético.

3.2.3 Corte e marcação das janelas

O processo de corte e marcação é realizado de maneira conjunta, sendo assim o corte e a rotulação da janela é feito ao mesmo tempo. Para cada janela criada na imagem sem *label* é criada uma janela análoga à imagem com *label*. Na janela criada na imagem com *label* é feita uma análise verificando qual a porcentagem de verde (marcação feita pelo radiologista para indicar presença de DCF)

presente nesta janela; caso a janela tenha uma porcentagem superior a 20% essa janela é considerada com DCF presente. Então a janela na imagem sem *label* é cortada e marcada com DCF presente nela.

Neste estudo também é adicionado uma característica para controlar o passo na qual a sobreposição é feita nas imagens, sendo esta característica chamada de *stride*. O *stride* tem como objetivo aumentar o número final de janelas geradas e deixar a localização da DCF mais precisa nas imagens. Isso é feito através do controle do passo do corte.

Sem aplicação desta técnica ao realizar o corte de uma janela, o algoritmo de criação da base passaria para janela seguinte respeitando as dimensões do corte feito para janela anterior, ou seja, sem sobreposição. Logo dessa maneira duas janelas jamais conteriam uma mesma região da imagem original. O *stride* possibilita cortar uma janela a partir de um valor contido na janela anterior. A figura 7 mostra a aplicação do *stride* para os cortes realizados na horizontal, a cor vermelha representa o corte atual, e a cor amarela o próximo corte que é realizado.

Processo de corte com Stride

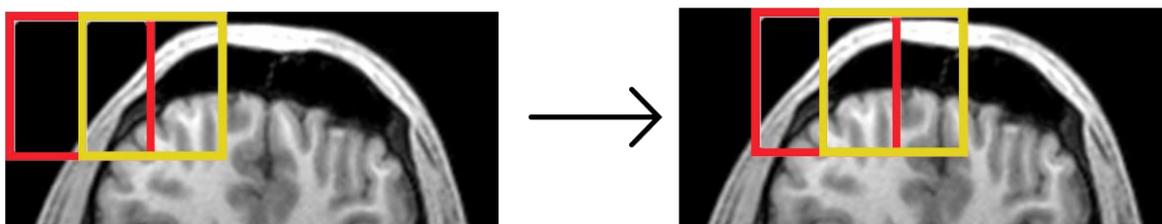


Figura 6: Ilustração do processo de stride. Fonte: Autor

3.2.4 Seleção de Região

Como pode ser observado no exemplo da Figura 5 (A) existem muitas janelas que não são relevantes para serem utilizadas na rede. Essas janelas são aquelas apenas com pixels de cor preta e as de rodapé presentes no canto inferior esquerdo de todas as imagens da base. Logo é feita uma verificação antes de

salvar quaisquer janelas para garantir que ela não se encaixa em nenhuma das duas situações descritas.

3.2.5 Aumento da Base de Dados

Em geral, o uso de Redes Neurais é utilizado em bases de dados com grande quantidade de registros. Apesar do aumento da base de dados proporcionado pelo processo explicado acima, o volume de dados ainda é muito pequeno e o processo gera um novo problema, o desbalanceamento da entre classes pois existem mais janelas sem DCF.

A fim de resolver este problema e aumentar a base de dados é proposta a utilização do método de *data augmentation* nas imagens (janelas) com DCF. Este processo consiste em gerar novas imagens a partir das imagens existentes. Estas novas imagens são geradas através da aplicação de alguns processos de transformação das imagens originais. Neste projeto cada imagem gerada é submetida a uma rotação múltipla de 20 graus, podendo sofrer inversão horizontais e verticais.

A quantidade de imagens geradas a partir de uma única imagem nesse projeto é dinâmica devido à otimização do tamanho das janelas e do parâmetro que controla a sobreposição de janelas. Isto é, cada indivíduo do AG influencia diretamente na proporção de imagens com DCF na base. Logo é necessário calcular o multiplicador de aumento de dados artificiais (*augmentation*) que é aplicado nas imagens com DCF, visando sempre que a base atinja a proporção de aproximadamente 50% de cada uma das classes. O multiplicador é encontrado através do uso da seguinte formula:

$$porcentagemDFC = \frac{(n * QtdDFC) * 100}{(QtdBase + [(N - 1) * QtdDFC]} \quad (1)$$

O valor do parâmetro n é testado entre os 0 e 10, sendo escolhido é aquele que resultar na *porcentagemDFC* (porcentagem de janelas com DCF) mais próxima de 50%. Esse processo é repetido a cada indivíduo antes de ser aplicada a CNN, sendo assim, o valor de *porcentagemDFC* é escolhido por indivíduo. O valor de $QtdDFC$ é sempre a quantidade de imagens com DCF gerada pelo indivíduo. E o valor de $QtdBase$ é o valor total de imagens geradas

pelo indivíduo. Foram realizados testes preliminares para determinar o espaço de busca de n .

3.2.6 Treino, teste e validação

Afim de poder realizar uma melhor avaliação desse estudo, o conjunto de dados utilizado é separado em 3 grupos: treino, teste e validação. Para o grupo de validação é separado todas as imagens obtidas de 1 dos 7 exames disponibilizados inicialmente. Este exame é submetido apenas a técnica do janelamento nas suas imagens. As janelas destas imagens não são utilizadas no treinamento das CNNs e nem para cálculo do fitness das soluções no AG, sendo utilizada apenas no fim após já ter obtido um conjunto de hiperparâmetros pelo pelo Algoritmo Genético. Vale ressaltar que essa separação busca avaliar o método desenvolvido para simular uma situação real, na qual o modelo (CNN) treinado anteriormente é aplicado para classificar novas imagens.

O restante dos exames é utilizado para compor os grupos de treino e teste, onde 20% das imagens são utilizadas para teste e 80% para treino. Após a separação as imagens são submetidas ao processo de janelamento e aumento da quantidade de janelas (*augmentation*). Enfim então o grupo de treina é utilizado para treinamento da CNN e pelo AG na busca dos hiperparametros, e de teste para metrificar este processo.

3.3 Algoritmo Genético

3.3.2 Algoritmos genéticos e as janelas

No trabalho anterior a esse (SILVA, 2019) foi constatado um possível viés no resultado final apresentado. No projeto as imagens da base foram divididas em janelas, sendo a escolha do tamanho das janelas pré-definida. E essas janelas foram rotuladas de maneira a indicar se a janela tinha ou não a presença de DCF. Porém algumas janelas que foram rotuladas com DCF tinham uma porcentagem muito pequena de presença de DCF (exemplo na Figura 7). Janelas como essas citadas podem ocasionar um viés na classificação realizada pela rede, pois são muito similares a janelas sem DCF.

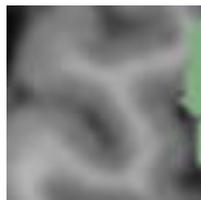


Figura 7: Janela criada com DCF presente sinalizada pelo label. Fonte: Autor

Sendo assim neste trabalho de mestrado, as dimensões da janela (altura e largura do retângulo) serão tratadas como parte dos parâmetros a serem otimizados pelo AG. Ou seja, genes do cromossomo do AG são responsáveis pela definição do tamanho das janelas. Os tamanhos das janelas (que são quadradas) são definidos por números inteiros (genes) entre 20 e 70. Os valores são baseados em experimentos realizados no trabalho anterior (SILVA, 2019) onde valores inferiores ou superiores a esses ocasionaram problemas na classificação realizada pela CNN.

3.3.3 *Stride*

Neste projeto a escolha do valor de *stride* é baseado na porcentagem de sobreposição entre janelas subsequentes, isto é, se configurado como 50% o valor de *stride* a próxima janela a ser cortada conterá 50% dos pixels da janela atual. Em testes iniciais, porcentagens maiores resultaram em menos imagens

com menos regiões similares e porcentagem menores o oposto. Da mesma forma que o tamanho das janelas, o valor de *stride* é definido como um parâmetro a ser otimizado pelo AG. Neste estudo o *stride* tem valor mínimo de 1 e máximo de 3.

3.3.4 Algoritmos genéticos e os hiper-parâmetros da CNN

Um dos fatores mais importantes para um bom resultado de uma rede é a utilização de bons hiperparâmetros que definem sua arquitetura e seu treinamento. Hiperparâmetros podem ser escolhidos por meio de testes exaustivos em busca as melhores combinações, por exemplo no método de *exhaustive grid Search* (FEURER, 2019). Outro modo de encontrar os parâmetros ideais é a utilização de meta-heurísticas para otimização, como os Algoritmos Genéticos.

Aqui, além de otimizar os parâmetros que definem as janelas aplicadas na imagem, o AG também otimiza alguns dos hiperparâmetros da rede. Serão codificados no cromossomo do AG alguns dos hiperparâmetros da CNN que mais impactam seu desempenho. Por exemplo, alguns elementos do AG codificam a arquitetura da CNN. Entre os hiperparâmetros que definem a arquitetura, são considerados aqueles que alteram o número de camadas convolucionais, número de neurônios nas camadas convolucionais e o número de neurônios nas camadas densas. A otimização dos hiperparâmetros do AG, em conjunto com outras características do problema de classificação, foi tratada recentemente em trabalho do nosso grupo de pesquisa (DEL-LAMA et al., 2020).

3.3.5 Indivíduo

Indivíduos do AG codificam em um vetor (cromossomo) as variáveis de decisão do problema de otimização. As variáveis de decisão do problema são aqueles parâmetros que queremos otimizar; aqui, estes são os parâmetros que definem o tamanho da janela, a sobreposição de janelas e alguns hiperparâmetros da CNN. Para cada variável de decisão, é necessário definir como ela é codificada no cromossomo do indivíduo. Aqui, são utilizados conjuntos de valores discretos pré-definidos para cada variável de decisão. Ou seja, as variáveis de decisão são representadas por valores discretos. Por exemplo, no caso do tamanho das

janelas, poderão ser utilizados valores inteiros entre 20 e 70 pixels. A figura 10 ilustra o vetor de características que inicialmente é utilizado neste projeto.

| | | | | | |
|---------|--------|--------|-------------------|---------------------------|--------------------------|
| Largura | Altura | Stride | N Camadas Ocultas | N Neurônios Camada Oculta | N Neurônios Camada Densa |
|---------|--------|--------|-------------------|---------------------------|--------------------------|

Figura 8: Ilustração do vetor de característica de um AG. Fonte: Autor

Sendo assim, é utilizado 20 indivíduos por geração, e o espaço de busca definido no projeto para cada indivíduo é de:

- **Largura:** 20 a 70 pixels
- **Altura:** 20 a 70 pixels
- **Stride:** 1 a 3, onde um apresenta 0% de cobertura, dois 50% de cobertura e três 33% de cobertura.
- **Número de Camadas Ocultas:** 1 a 3 camadas.
- **Número Neurônios Máximo Camadas Oculta:** 16 a 256 neurônios
- **Número Neurônios Camada Densa:** 16 a 256 neurônios.

O número de camadas camada oculta refere ao conjunto de uma camada convolucional com N neurônios e função de ativação ReLU, uma operação de *MaxPool* com *Kernel 2x2* e *dropout* de 0.2. O Número Neurônios Máximos Camada Oculta é o valor referente a última camada oculta presente na arquitetura apresentada pelos cromossomos do indivíduo, sendo assim as demais camadas terão valor decrescente a esse.

3.3.6 Função de fitness

Neste trabalho o AG tem como objetivo ajudar a encontrar os melhores parâmetros e hiperparâmetros para que a classificação da rede CNN seja a melhor possível, dado um número fixo de gerações. Esse valor máximo é resultante da função de avaliação, também conhecida como fitness.

A cada geração os indivíduos da população são submetidos a função de fitness. Através desse valor é possível aplicar algumas técnicas para selecionar indivíduos para a próxima geração; neste estudo é utilizado o elitismo e o torneio com operadores de seleção.

A princípio, a função de fitness escolhida para o projeto é a *f1-score*. Essa métrica é utilizada para avaliar a qualidade do modelo. A fórmula do *f1-score* é dada pela composição de 2 outras métricas extraídas da matriz de confusão. Dessa maneira é possível avaliar uma rede CNN através de um valor único. A *f1-score* é dada por:

$$f1 = \frac{2 * precisão * recall}{precisão + recall} \quad (2)$$

O valor da *f1-score* está compreendido entre 0 e 1, sendo quanto mais próximo de 1 melhor o modelo. Portanto neste estudo é buscado encontrar indivíduos que tenham notas mais próximas de 1. O grupo de dados no qual essa função é aplicada afim de otimizar é o grupo de teste, computando-se a *f1-score* para cada indivíduo presente, com as janelas e hiperparâmetros definidos pelo cromossomo do indivíduo, treinada utilizando-se o conjunto de treinamento. A Figura a seguir ilustra o uso do F1-Score.

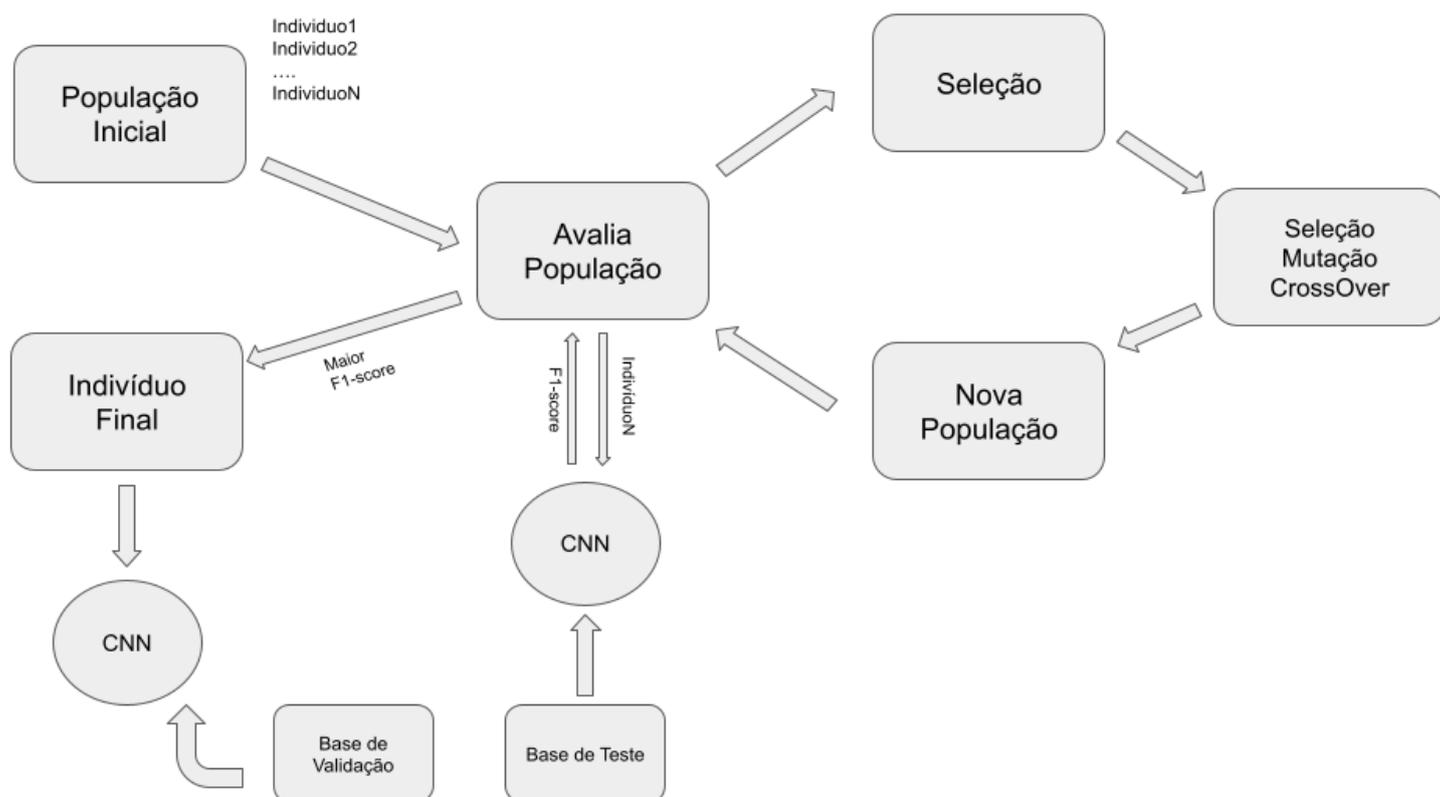


Figura 9: Figura ilustrando processo de avaliação do Algoritmo genético. Fonte: Autor

3.3.7 Elitismo e Torneio

Como dito anteriormente através da nota oferecida pela função de fitness é possível avaliar os indivíduos. Sendo assim, é possível selecionar indivíduos para reprodução. Primeiramente é aplicado o elitismo, que visa garantir que os indivíduos mais bem avaliados da população atual estejam presentes na população da próxima geração. Neste trabalho é aplicado o elitismo com valor 2, isto é, os dois indivíduos mais bem avaliados da população estão presentes na próxima população.

O segundo método aplicado é o do torneio, que consiste em fazer uma competição entre uma quantidade de indivíduos escolhidos aleatoriamente. O melhor indivíduo entre o grupo selecionado aleatoriamente, ou seja, aquele com melhor fitness, é escolhido para ser reproduzido. Neste estudo é aplicado o torneio entre 3 indivíduos selecionados aleatoriamente, sendo o vencedor do torneio é sempre aquele que tiver a maior nota de fitness entre os três

3.3.8 Mutação e Crossover

Por fim, o último passo para compor a população da próxima geração é aplicar operadores de transformação. Para aplicação destes processos primeiramente é selecionado dois pais com o método do torneio explicado anteriormente. Após a seleção feita é aplicado primeiro o processo de crossover e em seguida o processo de mutação.

O processo de mutação e crossover é repetido até que a população esteja completa. Neste trabalho, é utilizado a princípio o crossover de dois pontos. Para a mutação, é utilizada a mutação discreta uniforme, no qual quando a mutação é aplicada a um elemento do cromossomo, este é alterado aleatoriamente para um dos valores do subconjunto definido para gerar a população inicial. Por exemplo, caso o tamanho da janela seja mutado, o novo valor irá assumir um valor aleatório entre 20 e 70 pixels. Para o estudo foi definido como taxa de mutação o valor de 0.002 e como probabilidade de crossover 0.25.

3.4 Rede Convolutacional

Como explicado no capítulo anterior, o AG define a arquitetura da CNN. A cada novo indivíduo a ser executado é necessário reconfigurar a rede para as novas características de arquitetura.

Também é importante estabelecer que a CNN nesse projeto foi desenvolvida com o *framework PyTorch* (PASZKE et al, 2019). Este fato unido ao fato da rede ter uma nova arquitetura a cada indivíduo, gera um problema na relação entre a camada convolutacional e a camada densa, que mais a frente é explicado.

Outro ponto importante nesse estudo é o balanceamento da base, como explicado anteriormente, é aplicado um cálculo para gerar exemplos artificiais por meio de *augmentation* a fim de buscar sempre que a base fique o mais próximo possível de 50% de imagens para cada classe. Além desse tratamento, é calculada a proporção exata entre as classes pós *augmentation* e utilizar o este valor no método interno do *framework* para lidar com o desbalanceamento da base, assim buscando diminuir ao máximo esse problema.

3.4.1 A estrutura da camada convolutacional

Neste estudo é proposta uma arquitetura de rede onde a cada camada convolutacional o número de neurônios presentes nela deve dobrar. Portanto, na criação da camada oculta podemos estabelecer uma forte relação entre a quantidade de camadas e o número máximo de neurônios dentro dessa camada. Logo a cada novo indivíduo, é necessário a partir do número de camadas e o número máximo de neurônios calcular, se necessário, o número de neurônios das demais camadas. Por exemplo, um indivíduo com 3 camadas e com número máximo de 256 neurônios, ficaria a primeira camada com 64, segunda camada com 128 e a terceira camada com 256. A figura a seguir ilustra o exemplo:

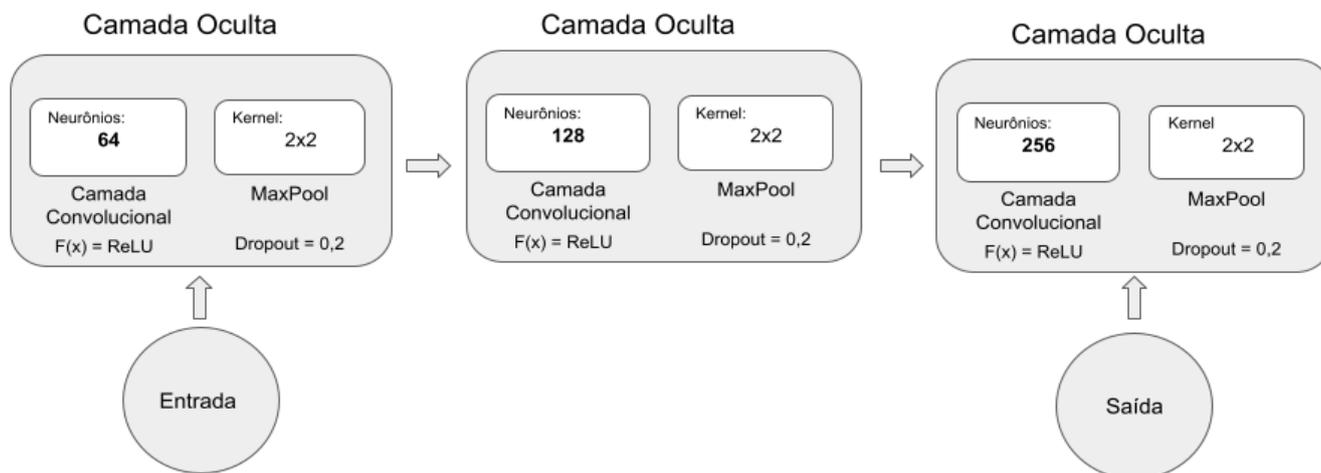


Figura 10: Ilustração representa uma rede convolucional com 3 camadas com 64,128, 256 neurônios por camada respectivamente. Todas com ReLU como função de ativação, *kernel* e *MaxPool* 2x2 e *dropout* de 0.2. Fonte: Autor

3.4.2 A relação da camada oculta com a camada densa

Como estabelecido no início deste tópico, existe uma relação criada entre a camada oculta, a camada densa e as características do indivíduo. Essa situação é criada pela codificação interna do pacote utilizado para desenvolvimento da CNN, o *PyTorch*.

No *framework* utilizado, quando a imagem utilizada tem dimensão distintas de largura e altura, isto é, não é uma imagem quadrada, ele não é capaz de transpor automaticamente a imagem da camada oculta para a camada densa. Internamente o *framework* não consegue calcular dado o tamanho das imagens e a arquitetura da rede a quantidade de operações de convolução resultantes e criar uma camada de maneira automática para fazer essa passagem entre as camadas.

Neste trabalho a cada novo indivíduo uma nova arquitetura e imagem com dimensões diferentes é apresentada à biblioteca, portanto é necessário

estabelecer o número de neurônios da camada densa de maneira explícita no código. O número de neurônios é obtido através da seguinte fórmula:

$$n_{neuronios} = n_{\max_{neuronios\ camada\ oculta}} * \left(\frac{altura}{2^{*n_{camadas}}}\right) * \left(\frac{largura}{2^{*n_{camadas}}}\right) \quad (3)$$

Assim com essa formula é possível a cada nova interação calcular o número correto de neurônios, para então funcionar a relação entre as camadas e o código funcionar como esperado.

3.4.3 Estrutura estática da rede

Apesar de termos alguns aspectos dinâmicos nessa rede, outros por sua vez são estáticos. Salvo os hiperparâmetros otimizados pelo AG, todos os outros da CNN são fixos, isto é, independente dos valores dados por um indivíduo do AG hiperparâmetros da CNN são definidos a priori.

Sendo assim, foi definido para esse estudo o valor de *dropout* de 0.2 para todas as camadas da rede (ocultas e convolucionais). Também para todas as camadas foi adotado a ReLU como função de ativação. Para o processo de *maxpooling* aplicado na camada de convolução, é utilizado um kernel de 2x2 com *stride* de 2. Todos estes hiperparâmetros foram definidos em testes preliminares.

3.5 Análise dos resultados

Como explicado anteriormente, a principal métrica utilizada para avaliar a rede é o *f1-score*, sendo esta obtida através de métricas extraídas da matriz de confusão.

A matriz de confusão traz como informação a classificação dos dados feito pelo modelo no conjunto de teste em comparação com sua classificação real. Os resultados obtidos nessa relação são classificados como (ver Figura 11):

- Falso Positivo (FP): Quantidade de janelas sem DCF que foram classificadas como com DCF.
- Falso Negativo (FN): Quantidade de janelas com DCF que foram classificadas como sem DCF.

- Verdadeiro Positivo (VP): Quantidade de janelas com DCF que foram classificadas como com DCF.
- Verdadeiro Negativo (VN): Quantidade de janelas sem DCF que foram classificadas como sem DCF.

| | | Valor Previsto | |
|------------------|----------|-----------------------|-----------------------|
| | | Positivo | Negativo |
| Valor Verdadeiro | Negativo | Verdadeiros Positivos | Falsos Negativos |
| | Positivo | Falsos Positivos | Verdadeiros Negativos |

Figura 11: Disposição das possíveis classificações de dados utilizando a matriz de confusão.

Fonte: Fonte: https://www.researchgate.net/figure/Figura-22-Matriz-de-confusao-dos-Sistemas-de-Deteccao-de-Intrusao_fig1_264741935

Baseados nessas classes é certo afirmar que os dados classificados corretamente estão presentes na diagonal principal da matriz, enquanto os classificados incorretamente estão fora dela. Através dessas relações presentes na matriz acima podemos calcular as métricas de avaliação utilizadas para obter o *f1-score*.

A precisão é dada por:

$$Precis\tilde{a}o = \frac{VP}{VP+FP} \quad (4)$$

O recall é dado por:

$$Recall = \frac{VP}{VP+FN} \quad (5)$$

Outra métrica também utilizada para analisar os resultados no projeto é a ROC. A ROC é uma curva de probabilidade capaz dizer o quanto o modelo consegue distinguir classes, assim como as demais ela é obtida através de métricas da matriz de confusão. Sendo as métricas utilizadas pela ROC, a sensibilidade apresentada a capacidade do modelo de acertar corretamente, e

especificidades utilizada para calcular a quantidade acertos do modelo. A sensibilidade é dada por:

$$\text{Sensibilidade} = \frac{VP}{VP+FN} \quad (6)$$

E especificidade por:

$$\text{Especificidade} = \frac{VN}{VP+FP} \quad (7)$$

A área sobre a curva ROC representa graficamente a qualidade do modelo desenvolvido. O gráfico é montado utilizando a 1- especificidade no eixo X e a sensibilidade no eixo Y. A figura abaixo é possível ver o gráfico da ROC.

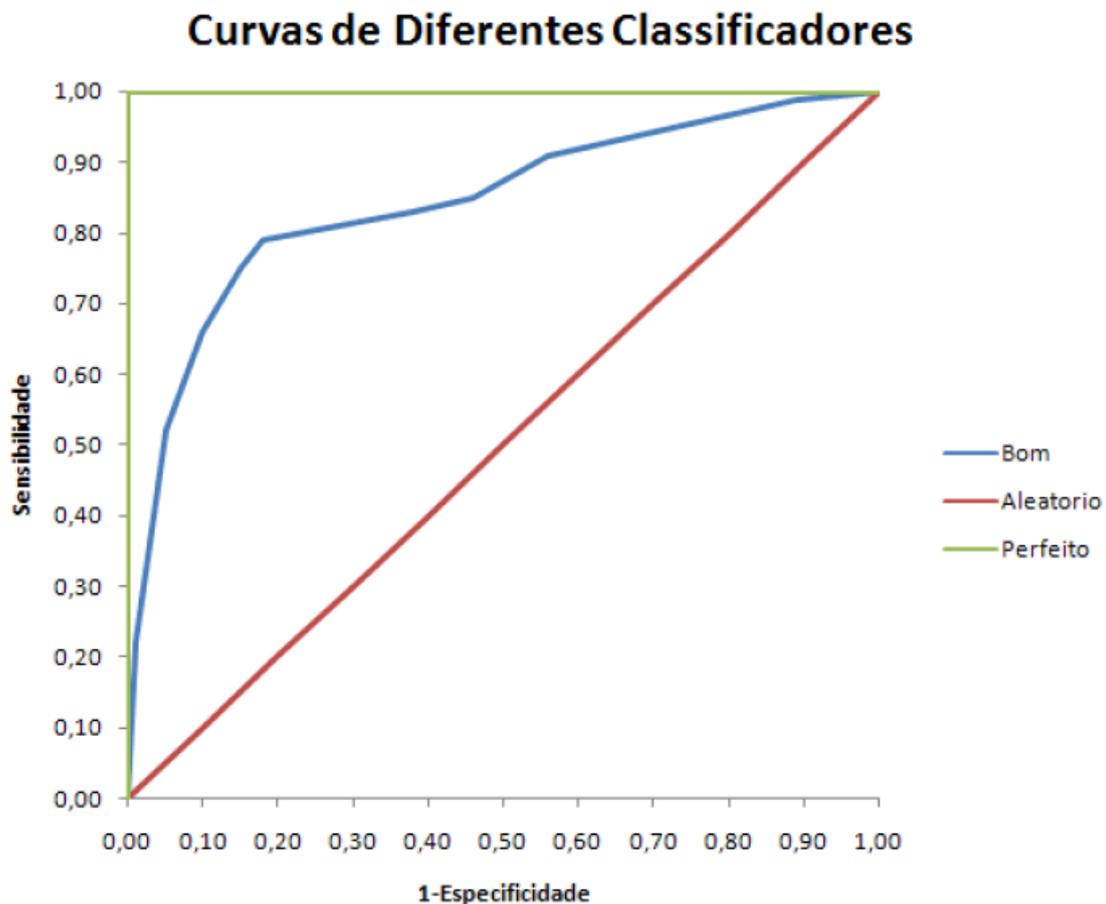


Figura 12: Grafico de Curva ROC.

Fonte: (SOUZA,2009)

3.6 Modelos Utilizados Para Comparação

A hipótese investigada aqui é que a CNN evoluída pelo AG e utilizando janelas deslizantes, também otimizadas pelo AG, produz resultados melhores de classificação de DCFs em pacientes com Epilepsia Refratária do que uma CNN com arquitetura definida a priori. Testa-se também a hipótese que tal modelo produz resultados melhores que uma rede neural que utiliza atributos extraídos das imagens. Assim, o método proposto foi comparado com uma rede CNN estática e uma rede MLP. Para ambos os modelos foram utilizadas janelas fixas e sem sobreposição. Os hiperparâmetros dos modelos foram definidos a priori. Para o MLP foi aplicado um extrator de características usando filtros pré-definidos, sendo utilizado para tal descritores de *Haralik*. Este descritor é capaz de descrever a imagem através de 14 características resultantes de 14 operações matemáticas diferentes. Os hiperparâmetros da CNN são os mesmos utilizados nos estudos (SILVA,2019; SILVA,2020).

No início deste projeto de mestrado foi desenvolvido um artigo onde foi feita uma comparação da rede CNN, onde não há pré-processamento, com um MLP que utiliza características pré-definidas nas entradas (SILVA, 2020). Em ambos os modelos, foram utilizadas janelas com tamanho pré-definido e sem sobreposição. Para a extração de características (filtros) para o MLP, foram utilizados os extratores de *Haralik*. O resultado obtido pela CNN foi apresentado em (SILVA, 2019).

4

Resultados

4.1 Comparação de modelos utilizando janelas fixas e sem sobreposição.

A seguir são apresentados os resultados da CNN estática e do MLP. Estes modelos serão comparados com o método proposto (CNN com janelas e hiperparâmetros otimizados pelo AG).

A CNN apresentada consta com a seguinte arquitetura 1 camada de convolução de 3x3 com 32 filtros, seguido por uma camada de *MaxPooling*, novamente uma camada de convolução de 3x3 com 62 filtros seguida por outra camada de *MaxPooling*. Por fim uma camada totalmente conectada com 128 neurônios com 50% de *dropout* e mais uma camada com apenas 1 neurônio responsável pela classificação, com a função de ativação a ReLU nas camadas densa e oculta, e sigmoide no neurônio de saída (SILVA,2019). Para o MLP foi proposta a arquitetura de 3 camadas com 40,120,40 neurônios respectivamente, taxa e aprendizado de 0.001, 400 épocas de treinamento, como otimizador foi usado o Adam e função de ativação a logística. (SILVA,2020)

A máquina utilizada nos projetos consta com a seguinte configuração: Notebook Dell modelo G3 3579, Intel Core i7-8750H CPU @ 2.20Ghz (12CPUs), 16GB RAM, Placa de vídeo NVIDIA GeForce GTX 1050Ti 4GB

4.1.1 Resultado da CNN Estática

A matriz de confusão apresentada na Tabela 1 mostra a média dos resultados da CNN para o conjunto de teste. O conjunto de teste é dado pelas janelas de uma imagem de paciente não utilizada no treinamento do modelo, sendo este fixo.

Tabela 1: Matriz de confusão obtida pela CNN

| | | Classificação | Classe Obtida | |
|-------------|---------|---------------|---------------|---------|
| | | | Com DCF | Sem DCF |
| Classe Real | Com DCF | 11 | 1 | |
| | Sem DCF | 4 | 8 | |

A acurácia obtida foi de 79%, a sensibilidade foi de 91% e a especificidade foi de 66%. Também foi calculada a área da curva ROC, que obteve uma cobertura de 79% como ilustra a figura 12.

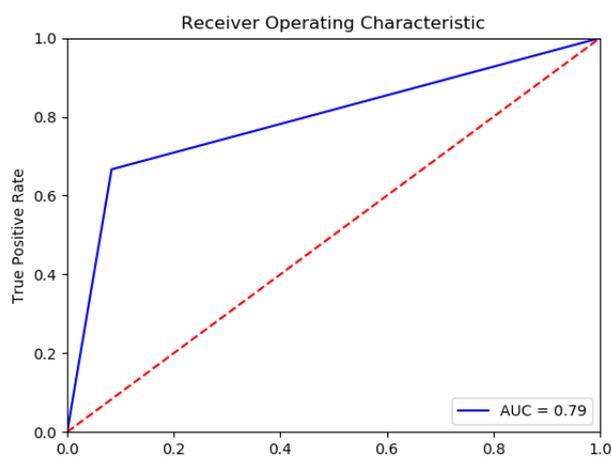


Figura 13: Curva ROC para CNN Fonte: Autor

4.1.2 Resultado MLP

A matriz de confusão apresentada na Tabela 2 mostra a média dos resultados obtidos pelo MLP. Também utiliza a mesma quantidade de janelas que o anterior.

Tabela 2: Matriz de Confusão obtida pelo MLP

| | | Classificação | Classe Obtida | |
|-------------|---------|---------------|---------------|---------|
| | | | Com DCF | Sem DCF |
| Classe Real | Com DCF | 10 | 2 | |
| | Sem DCF | 7 | 5 | |

A acurácia obtida foi de 62%, a sensibilidade foi de 83% e a especificidade foi de 41%. Também foi calculada a área da curva ROC, que obteve uma cobertura de 62% como ilustra a Figura 13.

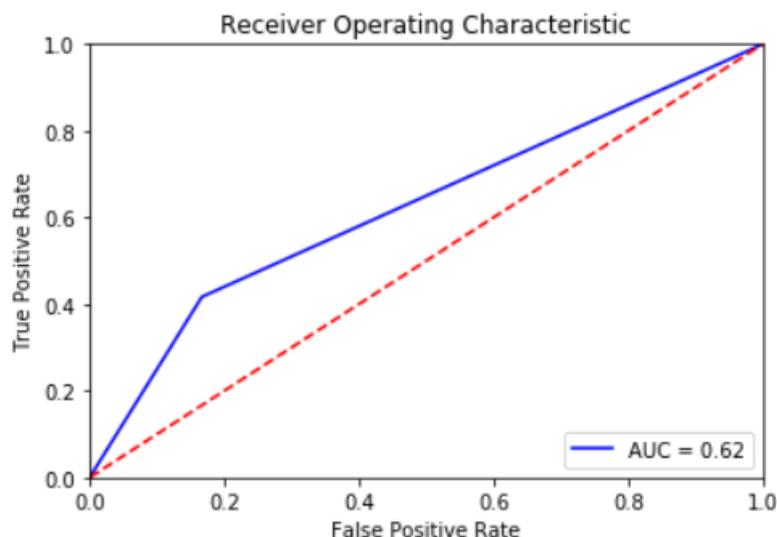


Figura 14: Representação da curva ROC. Fonte: Autor

4.2 Implementação do AG para otimização do tamanho das janelas

O AG foi inicialmente implementado, tendo sido realizado um experimento para otimizar a altura, largura e *stride* das janelas. Neste primeiro momento, os hiperparâmetros da CNN não foram otimizados.

Nesse experimento os valores e as metodologias usadas foram as mesmas que as explicadas no Capítulo 3. A premissa básica do experimento era encontrar os valores no qual a média de porcentagem presente nas janelas com

DCF seja máxima. Para esse intuito portanto foi necessário alterar a função de *fitness* do AG apenas para esse experimento. Aqui, a função de *fitness* é:

$$notaPopulacao = \frac{somaDcfPorcentagem}{QtdJanelasComDcf} \quad (8)$$

Como resultado da função de *fitness* é obtido a média de DCF nas janelas onde tem DCF presente, sendo essa a nota da população e quanto maior o resultado melhor.

Neste experimento uma população é composta por 25 indivíduos. A população inicial é criada de maneira aleatória, isto é, os indivíduos da população inicial são inicializados com características aleatórias dentro dos valores possíveis. A probabilidade de mutação utilizada é de 0.02 e a de crossover de 0.25. A Tabela 1 mostra a evolução do AG.

Tabela 3: Evolução do AG para otimização dos quadrantes

| Geração | Altura | Largura | Stride | Fitness |
|---------|--------|---------|--------|---------|
| 1 | 20 | 39 | 1 | 29.3 |
| 2 | 20 | 39 | 1 | 36.8 |
| 3 | 20 | 39 | 1 | 36.8 |
| 4 | 20 | 39 | 2 | 37.0 |
| 5 | 20 | 39 | 2 | 37.1 |
| 6 | 20 | 26 | 1 | 41.5 |
| 7 | 20 | 26 | 1 | 41.5 |
| 8 | 20 | 26 | 1 | 41.5 |
| 9 | 20 | 26 | 1 | 41.5 |
| 10 | 20 | 26 | 1 | 41.5 |
| 11 | 20 | 26 | 1 | 41.5 |
| 12 | 20 | 26 | 1 | 41.5 |
| 13 | 20 | 23 | 2 | 45.2 |
| 14 | 20 | 23 | 2 | 45.2 |
| 15 | 20 | 23 | 2 | 45.2 |
| 16 | 20 | 23 | 2 | 45.2 |
| 17 | 20 | 23 | 2 | 45.2 |
| 18 | 20 | 23 | 2 | 45.2 |
| 19 | 20 | 23 | 2 | 45.2 |
| 20 | 20 | 23 | 2 | 45.2 |

A figura 15 mostra de maneira visual a evolução da largura e altura do melhor indivíduo por geração do AG. Enquanto a figura 16 ilustra a evolução do valor de fitness destes indivíduos.

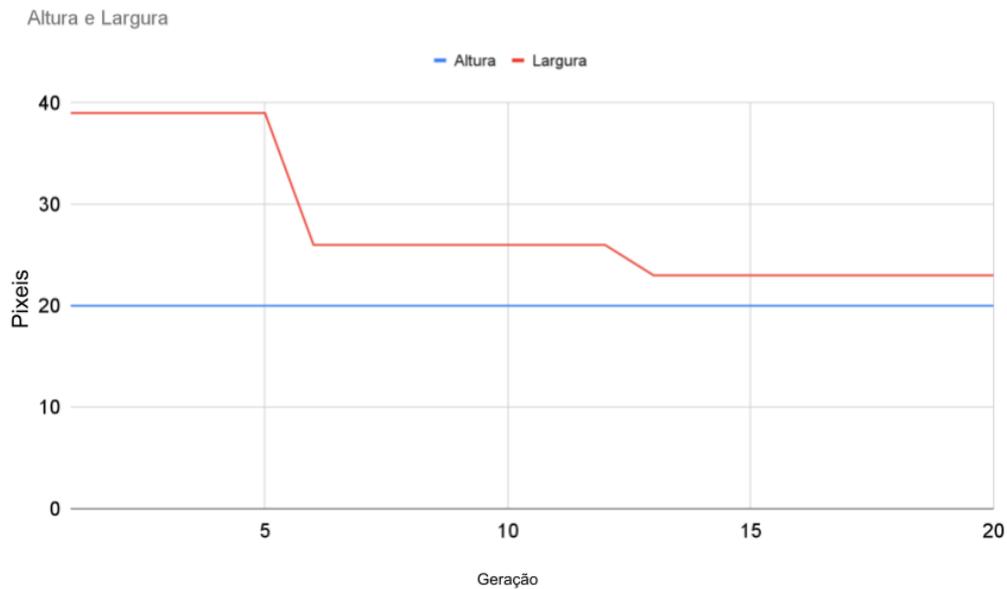


Figura 16: A figura ilustra a evolução dos valores de largura e altura otimizados pelo AG. Fonte: Autor

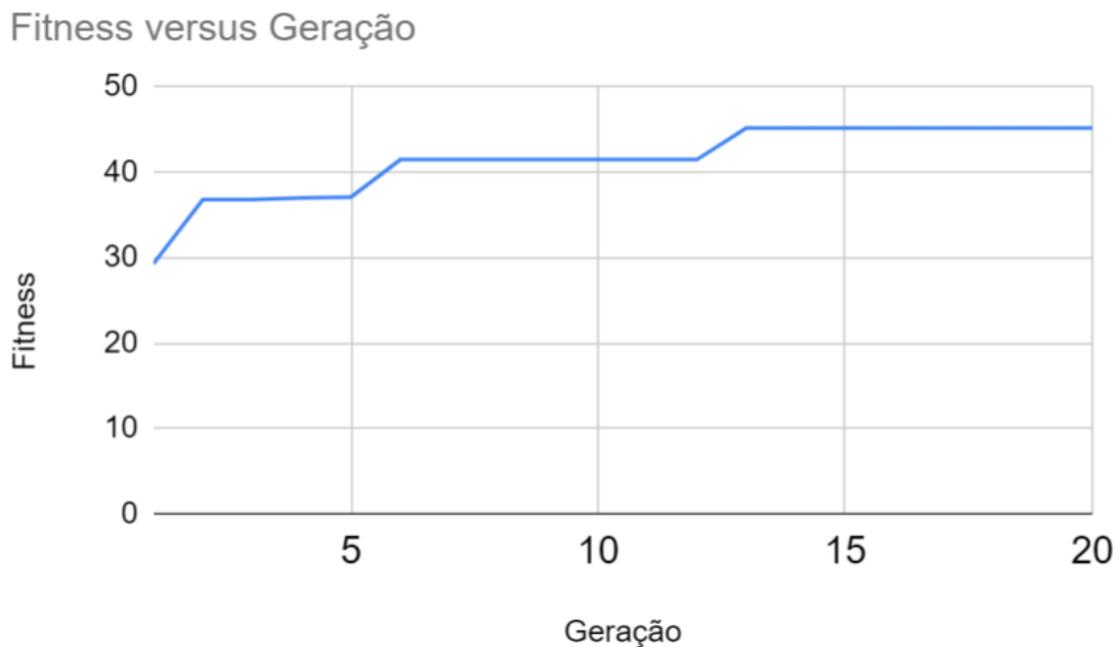


Figura 15: Figura ilustra a evolução do fitness ao longo das gerações. Fonte: Autor

4.3 Resultado do Modelo Proposto (CNN e janelas otimizadas pelo AG)

Devido ao alto custo computacional de se treinar e avaliar diversas CNNs em cada execução, o AG é evoluído por 5 execuções do AG utilizado para otimização das janelas e da CNN por 5 gerações, levando em média 24h10min por execução. As tabelas 5 a 8 representam os melhores indivíduos por gerações com seus respectivos valores de *f1-score* e acurácia.

Tabela 4: Resultado AG na primeira execução

| Ger | Largura | Altura | Stride | N Camada Oculta | N Neu. Cam. Oculta | N Neu. Cam. Densa | F1-Score | Acuracia |
|-----|---------|--------|--------|-----------------|--------------------|-------------------|----------|----------|
| 1 | 24 | 57 | 2 | 2 | 128 | 32 | 0.889 | 0.894 |
| 2 | 24 | 57 | 2 | 2 | 128 | 32 | 0.913 | 0.918 |
| 3 | 24 | 25 | 2 | 2 | 128 | 32 | 0.911 | 0.921 |
| 4 | 24 | 25 | 2 | 2 | 128 | 32 | 0.923 | 0.932 |
| 5 | 24 | 25 | 2 | 2 | 128 | 32 | 0.917 | 0.931 |

Tabela 5: Resultado AG na segunda execução

| Ger | Largura | Altura | Stride | N Camada Oculta | N Neu. Cam. Oculta | N Neu. Cam. Densa | F1-Score | Acuracia |
|-----|---------|--------|--------|-----------------|--------------------|-------------------|----------|----------|
| 1 | 22 | 47 | 2 | 2 | 128 | 256 | 0.871 | 0.892 |
| 2 | 68 | 24 | 2 | 2 | 128 | 32 | 0.897 | 0.897 |
| 3 | 46 | 24 | 2 | 2 | 128 | 16 | 0.905 | 0.911 |
| 4 | 46 | 24 | 2 | 2 | 64 | 16 | 0.913 | 0.919 |
| 5 | 22 | 24 | 2 | 2 | 128 | 16 | 0.917 | 0.933 |

Tabela 6: Resultado AG na terceira execução

| Ger | Largura | Altura | Stride | N Camada Oculta | N Neu. Cam. Oculta | N Neu. Cam. Densa | F1-Score | Acuracia |
|-----|---------|--------|--------|-----------------|--------------------|-------------------|----------|----------|
| 1 | 40 | 65 | 2 | 2 | 128 | 16 | 0.852 | 0.832 |
| 2 | 33 | 36 | 2 | 2 | 128 | 16 | 0.897 | 0.900 |
| 3 | 33 | 36 | 2 | 2 | 128 | 16 | 0.900 | 0.900 |
| 4 | 33 | 36 | 2 | 2 | 128 | 16 | 0.907 | 0.905 |
| 5 | 33 | 36 | 2 | 2 | 128 | 16 | 0.909 | 0.933 |

Tabela 7: Resultado AG na quarta execução

| Ger | Largura | Altura | Stride | N Camada Oculta | N Neu. Cam. Oculta | N Neu. Cam. Densa | F1-Score | Acuracia |
|-----|---------|--------|--------|-----------------|--------------------|-------------------|----------|----------|
| 1 | 50 | 20 | 2 | 1 | 256 | 64 | 0.834 | 0.830 |
| 2 | 63 | 30 | 2 | 2 | 64 | 16 | 0.856 | 0.843 |
| 3 | 36 | 30 | 2 | 2 | 64 | 16 | 0.891 | 0.903 |
| 4 | 36 | 30 | 2 | 2 | 256 | 16 | 0.88 | 0.886 |
| 5 | 36 | 30 | 2 | 2 | 64 | 16 | 0.900 | 0.933 |

Tabela 8: Resultado AG na quinta execução

| Ger | Largura | Altura | Stride | N Camada Oculta | N Neu. Cam. Oculta | N Neu. Cam. Densa | F1-Score | Acuracia |
|-----|---------|--------|--------|-----------------|--------------------|-------------------|----------|----------|
| 1 | 62 | 68 | 1 | 1 | 32 | 64 | 0.856 | 0.864 |
| 2 | 62 | 68 | 1 | 1 | 32 | 64 | 0.846 | 0.856 |
| 3 | 61 | 68 | 2 | 2 | 64 | 64 | 0.863 | 0.866 |
| 4 | 62 | 68 | 1 | 1 | 32 | 64 | 0.838 | 0.813 |
| 5 | 62 | 68 | 1 | 1 | 128 | 64 | 0.889 | 0.933 |

A figura 17 ilustra a evolução de largura e altura dos valores médios das cinco execuções por geração.

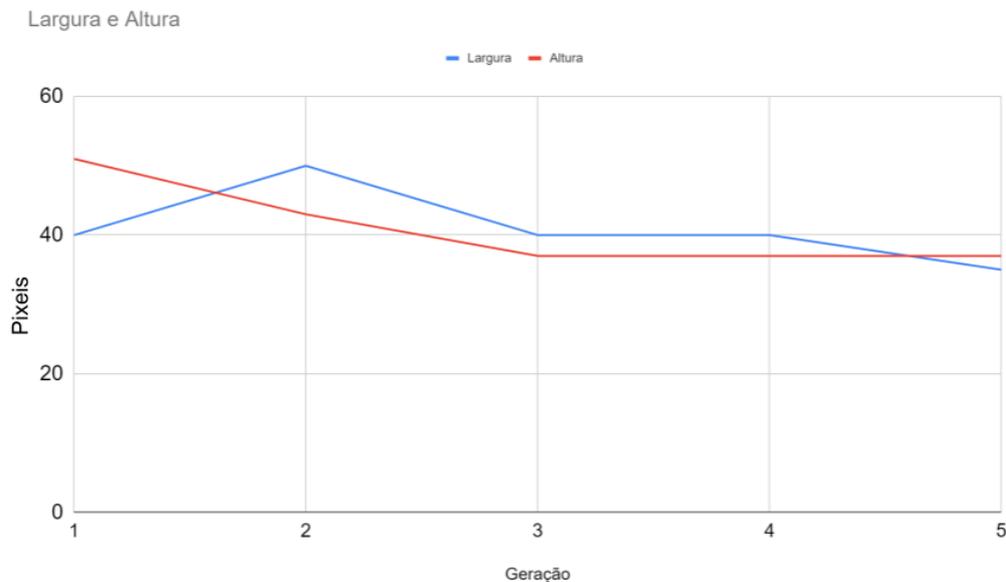


Figura 17: Ilustração da evolução dos valores médios de largura e altura das 5 execuções. Fonte: Autor

A figura 18 ilustra a evolução do número de neurônios da camada oculta juntamente com o número de neurônios da camada densa dos valores médios das cinco execuções por geração.

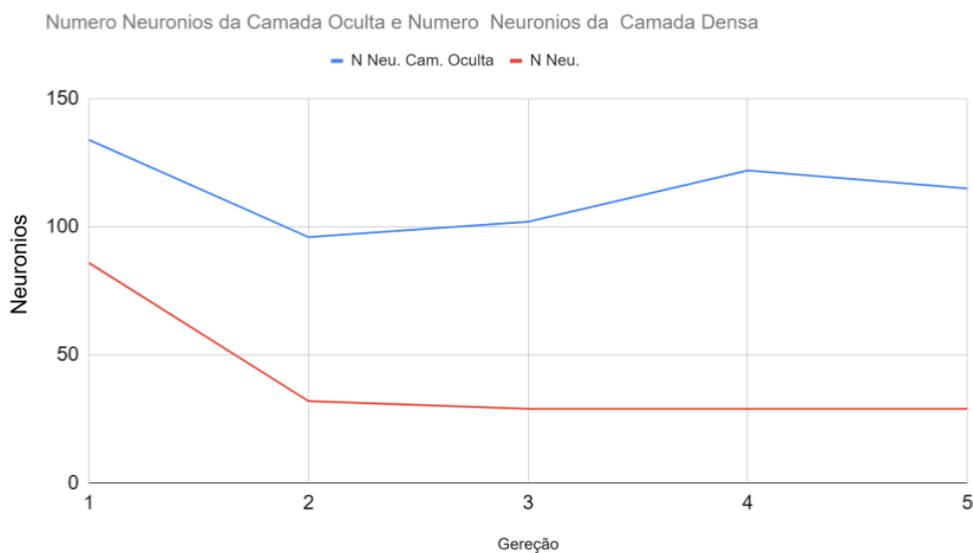


Figura 18: A Figura ilustra evolução do número de neurônios da camada oculta juntamente com o número de neurônios da camada densa dos valores médios das 5 execuções. Fonte: Autor

A figura 19 ilustra a evolução de fitness e acurácia dos valores médio das cinco execuções por geração.

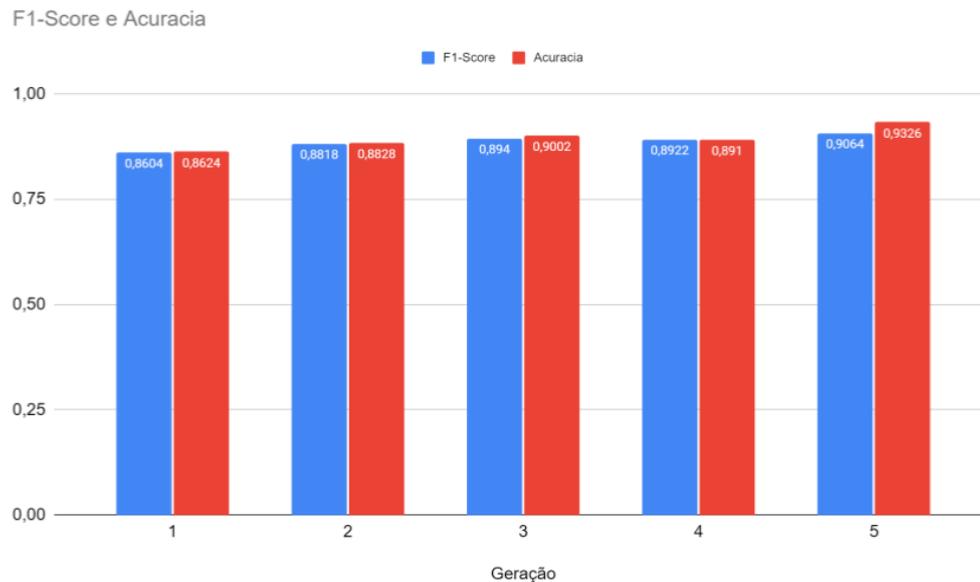


Figura 19: A figura 19 ilustra a evolução de fitness e acurácia dos valores médio das cinco execuções. Fonte: Autor

Dentre os 5 experimentos realizados o valor de f1-score mais alto foi de 0.917, sendo obtido nos experimentos 1 e 2 por dois indivíduos distintos, porém muito semelhantes. Em ambos foram comuns as características de Numero de Neurônios da Camada Oculta, Numero de camadas Ocultas, Stride; sendo respectivamente 128,2,2 os valores. O indivíduo da primeira execução tem largura e altura de 24x25, e o indivíduo da segunda execução de 22x24. O número de neurônios da camada densa também diverge entre eles, onde no primeiro experimento é de 32 e o segundo de 16. A figura 17 ilustra a arquitetura

utilizada pela CNN estática, enquanto a figura 18 ilustra a arquitetura do melhor indivíduo otimizado pelo AG.

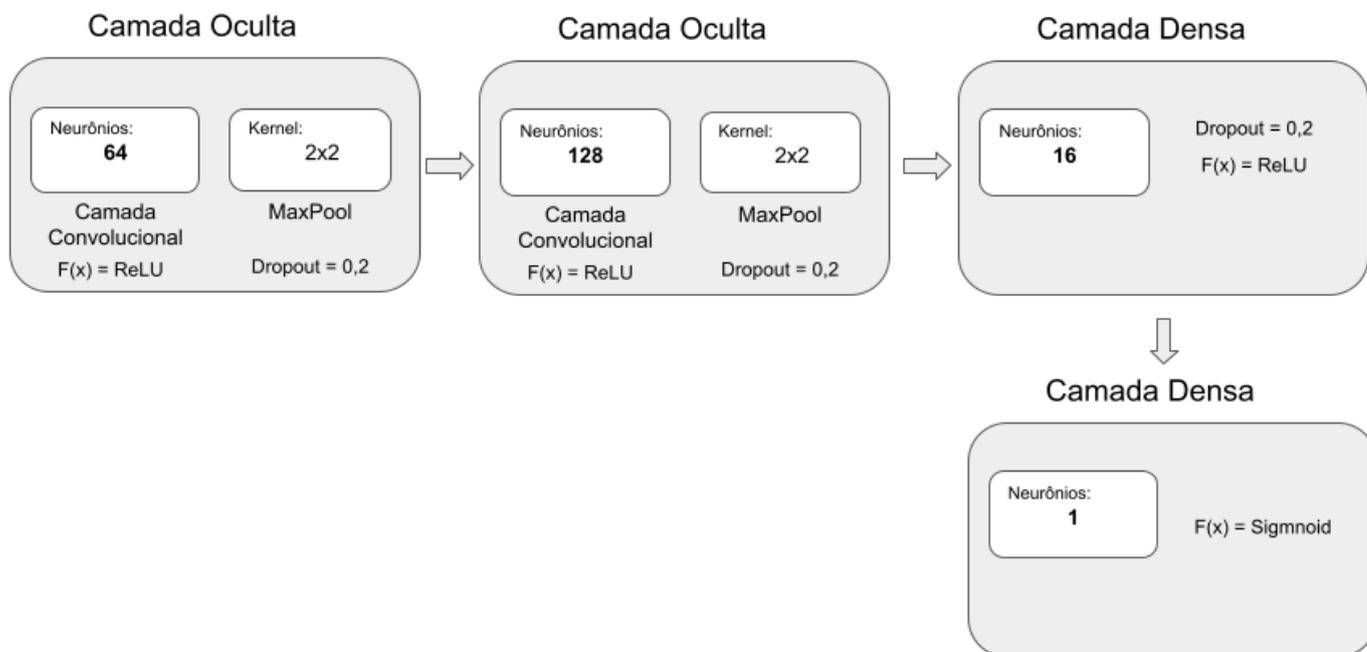


Figura 21: Arquitetura da CNN estática.

Fonte: Autor

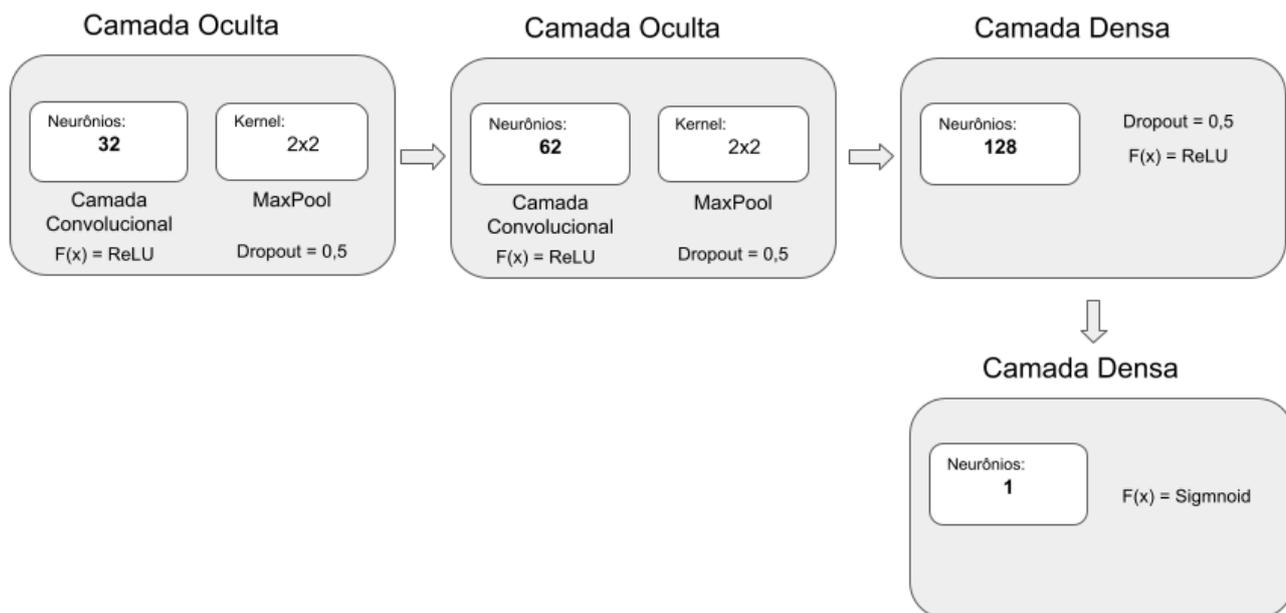


Figura 20: Arquitetura da CNN otimizada pelo AG.

Fonte: Autor

Também dentro das 5 execuções nota-se algumas prevalências, o valor de acurácia obtido foi o mesmo em 4 deles, sendo 0.933 de acurácia. O valor de *Stride*, Numero de Neurônios da Camada oculta e Numero de camadas ocultas é o mesmo também em 4 das 5 execuções realizados. A figura 22 é um comparativo dos valores de *f1-score* e acurácia entre o MLP, a CNN estática, o melhor individuo das cinco execuções da CNN com o AG e o resultado da média das 5 execuções.

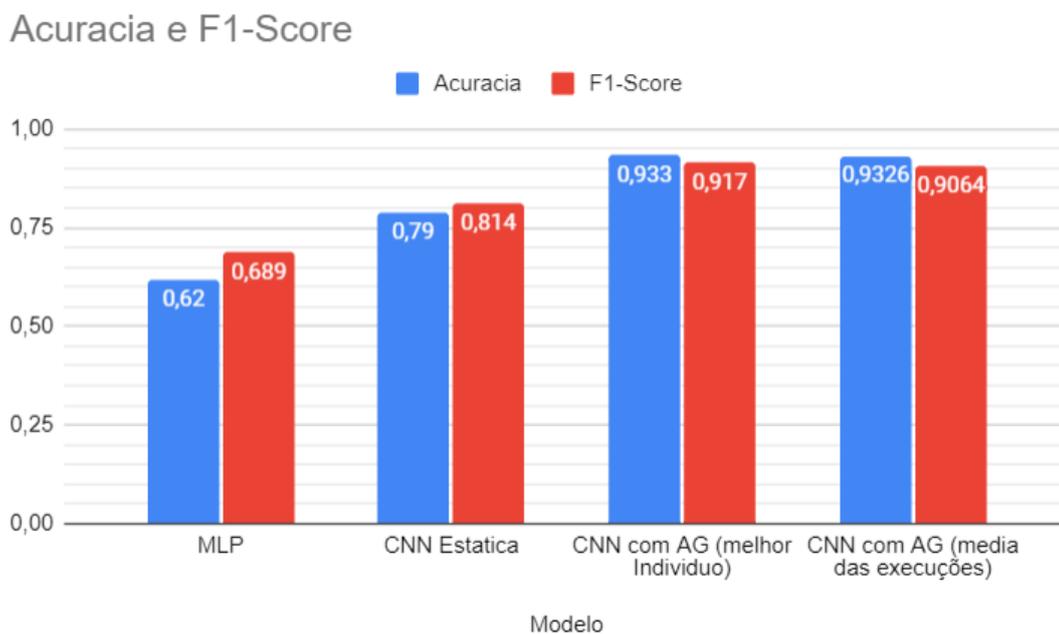


Figura 22: comparativo dos valores de *f1-score* e acurácia entre o MLP, a CNN estática, o melhor individuo das cinco execuções da CNN com o AG e o resultado da média das 5 execuções. Fonte: Autor

Vale ressaltar que se opta por utilizar uma imagem inteira para avaliar o método desenvolvido para simular uma situação real, na qual o modelo (CNN) treinado anteriormente é aplicado para classificar novas imagens.

4.3 Discussão

O resultado obtido no experimento preliminar com o AG é bastante interessante, considerando o comportamento da evolução olhando apenas para altura, largura e *stride* das janelas. Os experimentos mostraram que é possível obter uma cobertura bem alta de DCF nas imagens com DCF. A adição do *stride* para criação da base foi uma escolha positiva pois o AG evoluiu para um valor final onde o *stride* foi maior que 1.

A otimização realizada pelo AG trouxe resultados bastante satisfatórios para esse estudo, nota-se que em todas das execuções o indivíduo final proposto é um *f1-score* igual ou maior que 90%. Nota-se também uma prevalência muito forte na característica de *stride* e se mantendo igual na maioria dos melhores indivíduos durante as gerações. Além disso, podemos notar também a repetição e prevalência de algumas características no melhor indivíduo das execuções. Nos valores de largura e altura é possível notar que tenderam a ser valores próximos entre si apesar e haver a possibilidade de serem valores muito distintos, dessa maneira nota-se que nas 5 execuções janelas mais quadrangulares tiveram resultados melhores. Outro ponto importante a ser visto é tendência da rede da camada densa de ter um valor baixo de neurônios, enquanto a camada oculta concentra o valor de neurônios em 128.

Em geral nas 5 execuções, na maioria das características otimizadas, tenderam a ter os mesmos valores ou valores próximos em seus melhores indivíduos. Um experimento com um número maior de gerações pode indicar a predominância exata de algum valor de cromossomo no indivíduo.

É possível notar também que os melhores resultados obtidos pelo AG no experimento com a CNN tem uma forte relação entre os melhores resultados obtidos pelo AG para otimização do tamanho das janelas. Quanto mais próximo os valores de altura e largura entre a CNN otimizada e o valor resultante do experimento teste do AG, maior é o *f1-score* obtido. Assim podemos ver que um tamanho ótimo para os quadrantes, onde a porcentagem de DCF neles seja máxima, é benéfica para um resultado otimizado da rede.

No experimento com janelas fixas e sem sobreposição é possível observar que a escolha de uma CNN para classificação de DCF é mais interessante do que métodos convencionais, tendo uma acurácia e especificidade bem superiores ao MLP proposto com a técnica extração de características utilizadas nele. Na CNN, a seleção de filtros (características) é feita de maneira automática, não sendo necessária a extração de filtros pré-definidos.

Nos estudos anteriores (SILVA,2019) foi utilizado a acurácia como métrica de qualidade do modelo, comparando então a CNN otimizado com AG com os demais métodos, vemos um ganho significativo. Todos os melhores indivíduos adotados pelos experimentos do AG foram superiores aos demais estudos realizados, tendo o melhor deles um ganho superior a 10% na acurácia da classificação. Outro benefício da CNN otimizada é a possibilidade de gerar arquiteturas menores com qualidade igual ou melhor; CNNs menores implicam em velocidades de processamento e treinamento melhores, além de menor consumo de energia

4.5 Limitações

Uma das limitações encontradas nesse estudo foi o tempo de processamento, em média cada execução do AG levou cerca de 24h10min para ser concluído, dificultando assim a experimentação. Portanto estudos com computadores mais potentes ou número maior de gerações podem ser interessantes. A máquina utilizada no estudo consta com a seguinte configuração: Notebook Dell modelo G3 3579, Intel Core i7-8750H CPU @ 2.20Ghz (12CPUs), 16GB RAM, Placa de vídeo NVIDIA GeForce GTX 1050Ti 4GB.

Outra limitação também é quantidade de amostras iniciais disponíveis para o estudo. Idealmente a CNN tem resultados mais promissores com um volume de dados maior. Apesar deste estudo ter bons resultados, um número maior de amostras traria maior diversidade para rede e talvez até resultados melhores.

A resolução das imagens utilizadas nesse projeto também se mostrou um ponto limitante, uma vez que a quantidade de pixels presente das imagens

disponíveis acabou limitando o espaço de busca. Isto é, dado o tamanho das imagens se torna inviável um espaço de busca muito maior do que o apresentado nesse estudo, pois assim teríamos uma quantidade ainda menor de amostras para utilizar na CNN. Por sua vez esse problema acaba limitando também o espaço de busca proposto para arquitetura da camada oculta, pois como as imagens tem poucos pixels não é possível realizar muitas operações convolucionais.

5

Conclusão

Com base no estudo realizado, podemos inferir que o Algoritmo Genético traz resultados bastante benéficos para o projeto e para aplicação da CNN. Como levantado nas limitações do projeto anterior (SILVA, 2019) sobre o tamanho das janelas, temos que janelas e seu porcentual de DCF presente é um fator muito relevante para esse tipo de estudo utilizando a técnica do janelamento. Logo achar um número otimizado de largura e altura para as janelas trouxe um ganho para assertividade do modelo.

Podemos ver também que a sobreposição de janelas para criação da base foi bastante positiva, uma vez que a maioria das execuções os indivíduos com *stride 2* foram selecionados como os melhores ao fim. Foi possível também verificar algumas preferências dos parâmetros da CNN, como um número de camada densa e camada oculta de valores baixos dentro do espaço de busca proposto. Por sua vez a quantidade de neurônios da camada densa tendeu a ser um valor baixo e da camada oculta tendeu a ser alto dentro do espaço de busca. Com mais amostras, ou um conjunto de dados diferentes ou imagens com uma maior resolução, essa preferência pode ser outra.

A abordagem utilizada da otimização conjunta de parâmetros de rede junto com a composição da arquitetura da rede é promissora. Apesar da combinação de parâmetros os valores das janelas tendem dos melhores resultados tende a um valor próximo a otimização de porcentual de DCF máximo nas janelas. Outro estudo futuro relevantes é adicionar mais parâmetros ao AG para serem otimizados de maneira conjunta

Com base em todos os ganhos obtidos pelo AG, ficou nítido uma melhora na classificação quando se comparada com a CNN estica e MLP anteriormente desenvolvidos. A CNN estática já apresentava um ganho significativo em comparação ao MLP, porem seus parâmetros foram escolhidos através de teste manuais feito no projeto anterior. Com mais de 10% de melhoria na acurácia

entre as CNN, nota-se que o uso do AG buscar os melhores parâmetros para rede e o conjunto de dados propostos é muito benéfico a classificação. Assim, um estudo futuro considerando um espaço de busca maior destes parâmetros pode ser benéfico.

Em geral o uso de Algoritmos genéticos para otimização da seleção de quadrantes da técnica do janelamento, juntamente com a otimização da arquitetura e hiperparâmetros da Rede convolucionais para classificação e demarcação de Displasia Cortical Focal em Imagens de Ressonância Magnética é promissora. Os métodos propostos neste projeto são genéricos e podem ser aplicados em outros conjuntos de imagens para resolver mais problemas e imagens médicas. Um trabalho futuro é validar a técnica proposta em outras áreas de conhecimento. Neste projeto foi utilizado um número baixo de imagens, aplicar novamente a técnica para uma amostra maior de dados ainda é interessante para compreender melhor o comportamento da classificação, sendo este outro projeto futuro.

Referencias

GARZON, E. Epilepsia Refratária: Conceito e Contribuição das Novas Drogas Antiepilépticas e de outras Modalidades Terapêuticas. *Revista Neurociências*, [s. l.], v. 10, n. 2, p. 66–82, 2002.

WAN, P. A. K. The New England Journal of Medicine Early Identification Of Refractory Epilepsy. 2000.

FRENCH, J. A. *Refractory epilepsy: Clinical overview. Epilepsia*, v. 48, n. SUPPL. 1, p. 3–7, 2007. Disponível em: <https://doi.org/10.1111/j.1528-1167.2007.00992.x>

CAMARGO, D. P. C. et al. Displasia cortical focal, aspectos neurofisiológicos, imaginológicos e histológicos. *Journal of Epilepsy and Clinical Neurophysiology*, [s. l.], v. 16, n. 3, p. 106–111, 2010. Disponível em: <https://doi.org/10.1590/S1676-26492010000300005>.

KRSEK, P. et al. *Different features of histopathological subtypes of pediatric focal cortical dysplasia. Annals of Neurology: Official Journal of the American Neurological Association and the Child Neurology Society*, v. 63, n. 6, p. 758–769, 2008. Disponível em: <https://doi.org/10.1002/ana.21398>

AZEVEDO-MARQUES, P. M. D. Diagnóstico auxiliado por computador na radiologia. *Radiologia Brasileira*, [s. l.], v. 34, n. 5, p. 285–293, 2001. Disponível em: <https://doi.org/10.1590/s0100-39842001000500008>.

PALMINI, A et al. *Terminology and classification of the cortical dysplasias. Neurology* v. 62, n. 6 suppl 3, p. S2 LP-S8, 2004. Disponível em: <https://doi.org/10.1212/01.WNL.0000114507.30388.7E>.

SIMOZO, F. H. et. al. Caracterização e identificação de displasias corticais focais em pacientes com epilepsia refratária através de análise de imagens estruturais

de ressonância magnética nuclear. 2018. Universidade de São Paulo, Ribeirão Preto, 2018. Disponível em: <http://www.teses.usp.br/teses/disponiveis/59/59135/tde-13062018-140423/>.

WANG, H. et al. *Automated detection of focal cortical dysplasia using a deep convolutional neural network*. *Computerized Medical Imaging and Graphics*, v. 79, 2020. Disponível em: <https://doi.org/10.1016/j.compmedimag.2019.101662>.

DEV, KM B. et al. Automatic detection and localization of focal cortical dysplasia lesions in MRI using fully convolutional neural network. *Biomedical Signal Processing and Control*, v. 52, p. 218-225, 2019.

SILVA, S. H. “Uso de Redes Convolucionais para classificação de Displasias Corticais Focais em Pacientes com Epilepsia Refratária”. Trabalho de Conclusão do Curso de Bacharelado em Informática Biomédica, Universidade de São Paulo, Ribeirão Preto. 2019.

DEL LAMA, R. S. et al. (2022). Computer-Aided Diagnosis of Vertebral Compression Fractures Using Convolutional Neural Networks and Radiomics. *Journal of Digital Imaging*, 1-13.

RODRIGUES, A. L. et al. Seleção de Atributos para Apoio ao Diagnóstico do Câncer de Mama Usando Imagens Termográficas, Algoritmos Genéticos e Otimização por Enxame de Partículas. II Simpósio de Inovação em Engenharia Biomédica (SABIO 2018), Recife, Brazil, 2018.

SILVA, G. L. F. D. Diagnóstico de nódulos pulmonares em imagens de tomografia computadorizada usando redes neurais convolucionais evolutivas. 2017. 72 f. Dissertação (Programa de Pós-Graduação em Ciência da Computação) - Universidade Federal do Maranhão, São Luís, 2017. Disponível em: <http://tedeabc.ufma.br:8080/jspui/handle/tede/1534>.

CLEVERT, D. et al *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*. 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings, p. 1–14. Disponível em: <http://arxiv.org/abs/1511.07289>

LIMA, K. K. D. S. Desenvolvimento E Comparação De Redes Neurais Convolucionais Para Classificação De Objetos. Trabalho de Conclusão do

Curso, Universidade Federal Rural do Semi-Árido p. 1–12, 2019. Disponível em: <http://repositorio.ufersa.edu.br/bitstream/prefix/4316>.

ROJAS, R. The backpropagation algorithm. In: *Neural networks*. Springer, Berlin, Heidelberg, 1996. p. 149-182.

LINDEN, R. *Algoritmos genéticos (2a edição)*. Brasport, 2008.

EIBEN, A. E., & SMITH, J. E. (2003). *Introduction to evolutionary computing* (Vol. 53, p. 18). Berlin: springer.

FEURER, Matthias; HUTTER, Frank. Hyperparameter optimization. In: *Automated machine learning*. Springer, Cham, 2019. p. 3-33. Disponível em: https://doi.org/10.1007/978-3-030-05318-5_1

DEL-LAMA, R. S. *Algoritmos genéticos e redes neurais convolucionais para auxílio ao diagnóstico de fraturas vertebrais por compressão*. Dissertação (Mestrado em Computação Aplicada) - Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, *University of São Paulo*, Ribeirão Preto, 2020. doi:10.11606/D.59.2020.tde-07122020-220625.

PASZKE, A. et al., 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., pp. 8024–8035. Available at: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.

SILVA, S. H. et. al. Uso de Redes Neurais Convolucionais para Identificar Displasia Cortical Focal em Pacientes com Epilepsia Refratária. In: *Encontro Nacional De Inteligência Artificial E Computacional (ENIAC)*, 17. , 2020, Evento Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2020. p. 211-221. Disponível em: <https://doi.org/10.5753/eniac.2020.12130>.

SOUZA, C. *Análise de Poder Discriminativo Através de Curvas ROC*. 2009. Disponível em: <http://crsouza.com/2009/07/13/analise-de-poder-discriminativo-atraves-de-curvas-roc/>