
Diagnóstico de Alzheimer baseado em imagens de Ressonância Magnética utilizando Redes convolucionais

Davi Muller Suenari



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Monte Carmelo - MG
2025

Davi Muller Suenari

**Diagnóstico de Alzheimer baseado em imagens
de Ressonância Magnética utilizando Redes
convolucionais**

Trabalho de Conclusão de Curso apresentado à
Faculdade de Computação da Universidade Federal
de Uberlândia, Minas Gerais, como requisito exigido
parcial à obtenção do grau de Bacharel em Sistemas
de Informação.

Área de concentração: Sistemas de Informação

Orientador: Professor Daniel Stefany Duarte Caetano

Monte Carmelo - MG

2025

Dedico este trabalho à minha mãe, pelo apoio incondicional à minha trajetória acadêmica, à minha namorada, pelo suporte ao longo da faculdade, e a Deus, por me guiar e me conceder a oportunidade de chegar até aqui.

Agradecimentos

Agradeço, primeiramente, à Universidade Federal de Uberlândia (UFU) por proporcionar uma formação acadêmica de excelência e um ambiente de aprendizado enriquecedor, que contribuiu significativamente para meu crescimento profissional e pessoal.

Expresso minha profunda gratidão ao meu orientador, Professor Daniel Stefany Duarte Caetano, por sua dedicação, paciência e orientação ao longo do desenvolvimento deste trabalho. Seu conhecimento e incentivo foram fundamentais para a realização deste estudo.

A todos os professores, colegas de curso e amigos que, de alguma forma, contribuíram para essa trajetória, meu sincero obrigado.

Agradeço em especial minha companheira de jornada “Mayara” por todo apoio e suporte durante a jornada, nos momentos em que perdia a força ela me dava a mão para seguir a diante.

Por fim, agradeço à minha família e àqueles que sempre me apoiaram, oferecendo suporte e motivação para que eu pudesse chegar até aqui.

“O primeiro passo para a cura de uma doença é compreendê-la.”
(Paul Kalanithi)

Resumo

Nos últimos anos tem se observado a crescente prevalência de doenças como diabetes, doenças cardíacas e demência, assim como a doença de Alzheimer que se detectadas previamente, permite a adoção de diferentes procedimentos terapêuticos que reduzem a sua evolução. Diante dos desafios dos diagnósticos tradicionais, especialmente para o Alzheimer, estudos apontam que as análise de imagens Ressonância magnética (RM) e sinais do Eletroencefalograma (EEG) tem se apresentadas como alternativas, no entanto apresentam limitações quanto a sua eficiência. Porém, a Inteligência Artificial (IA) vem se destacando como uma ferramenta crucial na medicina aprimorando o diagnóstico e tratamento dessas doenças. Entretanto, vários são os desafios encontrados no aprimoramento da eficiência dos algoritmos aplicados, exigindo ajustes nos componentes estruturais dos algoritmos além de diferenciadas técnicas de pré-processamento, quando estas ultimas não permitem um aumento no reconhecimento dos padrões de Alzheimer. Diante disso, este trabalho propõe a análise dos parâmetros de acurácia, perda, número de camadas, épocas e taxa de aprendizado na arquitetura ResNet 3x3 e 5x5, aplicada a um banco de dados público de imagens de MRI. Como principal métrica de avaliação, utilizou-se a acurácia, alcançando um melhor resultado de 50% com a ResNet 5x5, utilizando 20 camadas, 100 épocas e uma taxa de aprendizado de 0.1.

Palavras-chave: Alzheimer, ResNet, Ressonância Magnética, Análise de parâmetros, base de dados pública.

Lista de ilustrações

Figura 1 – Exemplo de RM	16
Figura 2 – Exemplo de CNN	17

Lista de tabelas

Tabela 1 – Resumo Comparativo dos Trabalhos Relacionados	22
Tabela 2 – Resultados da Variação dos Parâmetros para Filtros Convolucionais 3x3	33
Tabela 3 – Resultados da Variação dos Parâmetros para Filtros Convolucionais 5x5	34

Lista de siglas

CNN Rede neural convolucional

DA Doença de Alzheimer

EEG Eletroencefalograma

FN Falsos negativos

FP Falsos positivos

IA Inteligência Artificial

IRM Imagens de Ressonancia Magnetica

LCR Líquido cefalorraquidiano

MPL Redes neurais profundas

OMS Organização Mundial de Saúde

ResNet Rede Neural Residual

RM Ressonância magnética

TN Verdadeiros negativos

TP Verdadeiros positivo

VGG Visual Geometry Group

VGG16 Visual Geometry Group - 16 layers

Sumário

1	INTRODUÇÃO	12
1.1	Motivação	12
1.2	Problema	13
1.3	Hipótese	13
1.4	Objetivos	13
1.4.1	Objetivo geral	13
1.4.2	Objetivos específicos	14
1.5	Contribuições	14
1.6	Organização da Monografia	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Fundamentação Teórica	15
2.1.1	Doença de Alzheimer e Formas de Reconhecimento	15
2.1.2	Fundamentos de Imagens de Ressonancia Magnetica (IRM)	15
2.1.3	Redes Neurais aplicadas no processamento de IRM	17
2.2	Trabalhos Relacionados	18
2.2.1	Detecção de Doença de Alzheimer a partir de Neuroimagens Utilizando Redes Neurais Convolucionais	18
2.2.2	Aprendizado de características para classificação de da doença de Alzhei- mer	19
2.2.3	Utilização de Redes Convolucionais Para Classificação e Diagnóstico da Doença de Alzheimer	20
2.2.4	Considerações finais	21
3	EXPERIMENTOS E ANÁLISE DOS RESULTADOS	23
3.1	Metodologia	23
3.1.1	Seleção da base de dados	24
3.1.2	Variação dos Parâmetros	24

3.1.3	Métodos para a Avaliação	25
3.2	Detalhes de Implementação	25
3.3	Resultados Alcançados	33
3.4	Discussão e Avaliação dos Resultados	34
4	CONCLUSÃO	37
4.1	Conclusão	37
4.2	Trabalhos Futuros	38
	REFERÊNCIAS	40

Introdução

Atualmente, algumas das doenças mais prevalentes em todo o mundo incluem diabetes, doenças cardíacas e demência conforme a Organização Mundial de Saúde (OMS), com a Doença de Alzheimer (DA) sendo uma das formas mais comuns de demência (ORGANIZATION et al., 2015). À medida que a tecnologia continua a evoluir, a área médica tem se beneficiado cada vez mais da integração de ferramentas tecnológicas avançadas, sendo a IA uma das mais notáveis. A IA, por meio da análise de vastos conjuntos de dados, demonstra a capacidade de fornecer avaliações e conclusões cruciais para diagnóstico e tratamento médico.

1.1 Motivação

A IA está desempenhando um papel significativo como uma ferramenta auxiliar no diagnóstico de várias doenças, incluindo o Alzheimer. Entre as diversas abordagens utilizadas para o diagnóstico, destaca-se o uso de RM, em que a IA analisa imagens para identificar padrões. Nessa modalidade de diagnóstico, foi alcançada uma notável precisão de 95% na classificação dos indivíduos (MAIA et al., 2019). Além dessa técnica, outra abordagem se vale do EEG, com a qual se obteve uma taxa de acerto de 91,5% (RODRIGUES, 2011). Os métodos mais indicados para o diagnóstico da DA, sem o auxílio da IA, incluem a análise do exame de sangue e o Líquido cefalorraquidiano (LCR) (VARESI et al., 2022). Embora o exame de sangue seja amplamente recomendado como um auxiliar no diagnóstico, enfrenta uma limitação significativa, que é a sua complexidade na detecção do Alzheimer nos estágios iniciais (DELABY; TEUNISSEN, 2023). Por outro lado, o exame do LCR é geralmente empregado apenas para identificar a causa da doença em si, sendo realizado após o diagnóstico já ter sido estabelecido (HAMPEL et al., 2023).

1.2 Problema

A DA figura entre as demências mais prevalentes na sociedade atual (World Health Organization, 2023). Sua complexidade decorre, em grande parte, da ausência de uma cura definitiva, isto infelizmente resulta em muitos pacientes sem diagnóstico ou até mesmo diagnosticados erroneamente. Nos Estados Unidos, atualmente, estima-se que 5,4 milhões de cidadãos vivenciam essa doença, e é previsto um acréscimo de 1 milhão de novos casos até o ano de 2050 (ASSOCIATION et al., 2016). A demora ou imprecisão no diagnóstico acarreta inúmeros desafios para os pacientes, incluindo a falta de acesso a tratamentos paliativos e o acompanhamento médico adequado. Além disso, a doença pode desencadear problemas psicológicos significativos, como estresse, alterações de humor e ansiedade, particularmente quando o paciente não tem suas necessidades atendidas ou a interação com cuidadores ou familiares não é agradável. O impacto emocional, quando não abordado, agrava ainda mais a situação (KALES; GITLIN; LYKETSOS, 2015). Para identificar a doença, são empregados diversos métodos, incluindo análise de antecedentes e avaliação das funções cognitivas do paciente. No entanto, por meio desses métodos, não se pode afirmar com 100% de certeza que a causa seja o Alzheimer, devido a possíveis obstáculos, como o estágio da doença, distúrbios de comportamento ou outros fatores complicadores. Em resumo, mesmo após avaliações médicas, sem um exame comprobatório, o diagnóstico não pode ser estabelecido de forma definitiva (NITRINI et al., 2005). No entanto, a IA pode desempenhar um papel relevante na assistência ao diagnóstico, especialmente quando as análises médicas não são conclusivas. A precisão na interpretação de exames pela IA, como anteriormente mencionado, pode ser satisfatória, possibilitando a elucidação de dúvidas médicas ou análises imprecisas e, assim, contribuir para uma resposta diagnóstica mais precisa.

1.3 Hipótese

Neste trabalho trabalharemos as seguintes hipóteses:

- ❑ Qual o método mais efetivo?
- ❑ Quais melhorias a serem implementadas de forma que haja aumento no percentual de predição?

1.4 Objetivos

1.4.1 Objetivo geral

Este trabalho tem como objetivo a análise da influência dos principais parâmetros e seus impactos sobre o desempenho de uma rede neural a identificação e seleção das

melhores configurações para a obtenção de um bom desempenho na predição de Alzheimer.

1.4.2 Objetivos específicos

- ❑ Selecionar uma base de dados de imagens pública para os testes;
- ❑ Estudar os principais parâmetros envolvidos no bom desempenho de uma rede neural durante a classificação;
- ❑ Com base na arquitetura da rede neural definida, aferir o desempenho alcançado conforme a variação de parâmetros pré-determinados;
- ❑ Configurar a arquitetura da rede com os valores de parâmetros ótimos e verificar a eficiência da rede.

1.5 Contribuições

Ao se utilizar uma base de dados pública antes de se aplicar técnicas de pré-processamento elaboradas, com objetivo de aumentar o ganho de diferentes algoritmos de IA, é necessário descobrir qual a influência dos parâmetros básicos do algoritmo sobre a eficiência, para somente após explorar outras abordagens (SILVEIRA, 2019). Por esta razão, a contribuição deste trabalho está na geração e compartilhamento dos efeitos da alteração de tais parâmetros com a comunidade científica, gerando um ponto de partida para aplicação de outras abordagens que podem aumentar o nível no reconhecimento de padrões, evidenciando a configuração ótima, reduzindo esforços iniciais.

1.6 Organização da Monografia

A organização da monografia ficará:

- ❑ Capítulo 1 – Introdução e Objetivos.
- ❑ Capítulo 2 – Fundamentação Teórica e Trabalhos relacionados
- ❑ Capítulo 3 – Experimentos e Análise dos Resultados
- ❑ Capítulo 4 – Conclusões
- ❑ Capítulo 5 – Referências Bibliográficas.

Fundamentação Teórica e Trabalhos Relacionados

2.1 Fundamentação Teórica

2.1.1 Doença de Alzheimer e Formas de Reconhecimento

A doença de Alzheimer é uma condição neurodegenerativa que resulta em perda progressiva de habilidades motoras e de memória ao longo do tempo (World Health Organization, 2023). À medida que a doença avança, essas perdas se tornam mais evidentes e podem ser observadas a olho nu. Infelizmente, ainda não existe uma cura para essa doença, mas existem vários métodos de diagnóstico utilizados para detectá-la, que incluem (ASSOCIATION, 2019):

- ❑ Avaliação do histórico familiar da doença.
- ❑ Entrevistas com familiares para identificar mudanças comportamentais e motoras.
- ❑ Realização de testes cognitivos para avaliar a função mental e identificar deficiências.
- ❑ Exames de sangue para detectar desequilíbrios em substâncias relacionadas à doença.
- ❑ Uso de RM para identificar anomalias cerebrais que possam estar associadas à doença.

2.1.2 Fundamentos de IRM

2.1.2.1 RM e Alzheimer

Na área da medicina, a RM é uma ferramenta valiosa para a detecção da doença, inclusive em estágios precoces (FRISONI et al., 2022). Nesse exame, os pacientes são

submetidos a um procedimento no qual entram em um aparelho que emite ondas de rádio para obter imagens dos tecidos biológicos em diversas partes do corpo. No contexto do Alzheimer, os médicos utilizam as imagens da região craniana para identificar possíveis anomalias no cérebro que possam estar associadas à doença (JACK et al., 2023). Entre várias anomalias que podem ser identificadas, uma que pode se manifestar de maneira visualmente distinta é a perda de tecido na região cerebral (FRISONI et al., 2022). Na Figura 1, é possível observar a diferença nos tecidos registrados pela ressonância. A primeira ressonância apresenta a imagem de um cérebro saudável, enquanto a segunda exibe a imagem de um cérebro de um paciente com Alzheimer, destacando claramente a disparidade nos tecidos.

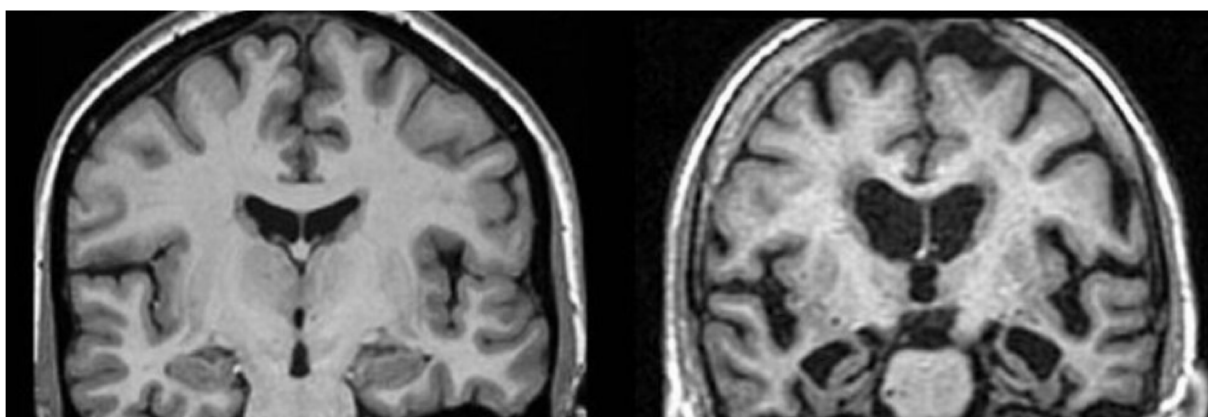


Figura 1 – Exemplo de RM

Fonte: [bbc.com/news/health-31807961](https://www.bbc.com/news/health-31807961)

2.1.2.2 Algoritmos de Processamento de IRM

Dado que a RM possui a capacidade de oferecer informações significativas sobre pacientes com Alzheimer, tem se tornado cada vez mais comum o emprego de tecnologias para uma compreensão mais profunda desses dados (RUDROFF; RAINIO; KLÉN, 2024). Nesse cenário, a utilização de algoritmos capazes de adquirir e processar imagens de RM com o propósito de gerar análises e conclusões acerca das informações coletadas está em destaque. Dentro desse âmbito, encontramos a aplicação de técnicas de IA, tais como:

- ❑ Machine Learning (Aprendizado de Máquina): Por meio de algoritmos de Machine Learning, é possível treinar modelos que podem identificar padrões nas imagens de RM, auxiliando no diagnóstico e no acompanhamento da progressão do Alzheimer (FOROUGHPOUR et al., 2023).
- ❑ Deep Learning (Aprendizado Profundo): O Deep Learning, uma subárea do Machine Learning, é empregado para análises mais complexas e automatizadas das imagens de RM, possibilitando uma avaliação ainda mais detalhada das informações e aprimorando o entendimento da doença (SALEHI et al., 2020).

2.1.3 Redes Neurais aplicadas no processamento de IRM

2.1.3.1 Redes Neurais Convolucionais (CNNs)

Uma Rede neural convolucional (CNN) é um algoritmo de aprendizado profundo desenvolvido para a classificação de imagens, composta por três componentes fundamentais: a camada convolutiva, camadas de pooling e funções de ativação (LECUN; BENGIO; HINTON, 2015).

As camadas convolutivas desempenham um papel crucial ao identificar padrões e características em dados de grade, como imagens, utilizando filtros que atuam como detectores de características (GOODFELLOW; BENGIO; COURVILLE, 2016), conforme exemplificado na Figura 2 pelas Conv Layer 1, 2, 3 e 4.

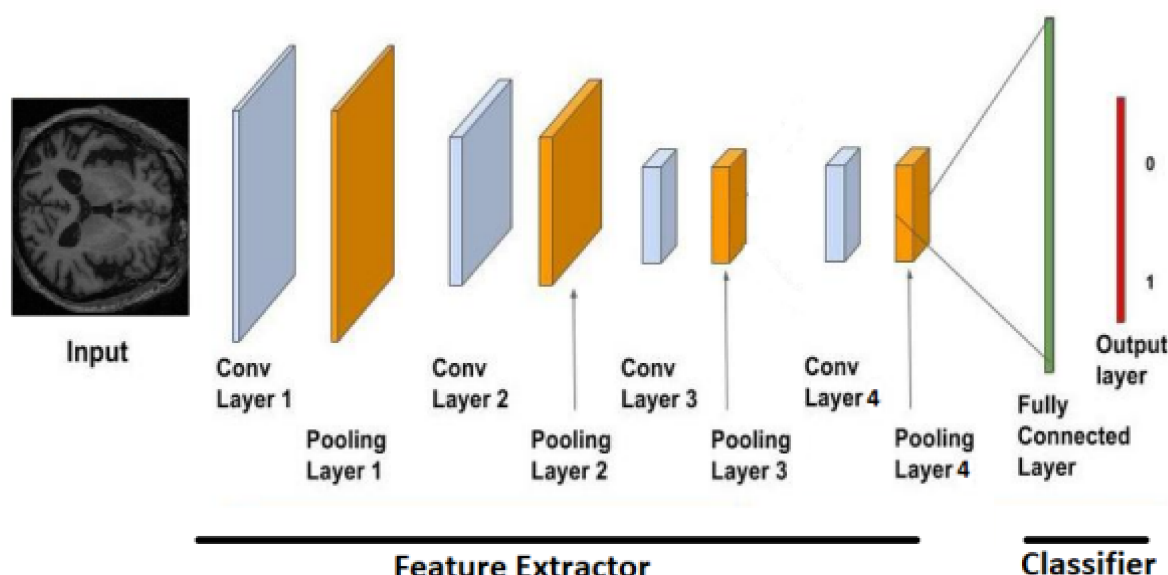


Figura 2 – Exemplo de CNN

Durante a convolução, esses filtros são aplicados a regiões menores da entrada, destacando padrões relevantes e gerando um mapa de características que ressalta elementos distintivos na imagem.

As camadas de pooling são responsáveis por reduzir a dimensionalidade espacial da saída das camadas convolutivas, preservando as características mais significativas e proporcionando uma representação mais condensada dos dados (KRIZHEVSKY; SUTSKER; HINTON, 2012), conforme exemplificado na Figura 2 pelos Pooling Layer 1, 2, 3 e 4. A aplicação de pooling envolve a utilização de uma janela sobre a saída da camada convolutiva, consolidando informações e reduzindo a resolução espacial de maneira eficiente.

As funções de ativação, como a ReLU, são amplamente utilizadas em redes neurais (NAIR; HINTON, 2010), pois introduzem não-linearidades nas CNN, permitindo que a rede aprenda padrões complexos e relações não lineares nos dados. Durante a propagação

para a frente, a função de ativação é aplicada aos resultados da convolução ou pooling, ativando neurônios positivos e zerando os negativos. Essa não-linearidade desempenha um papel crucial ao capacitar a rede a assimilar representações mais complexas e refinadas durante o processo de aprendizado.

Após essas fases, a CNN é capaz de categorizar imagens, embora, assim como o cérebro humano, alcance uma certeza inferior a 100% na distinção de determinadas imagens (LECUN; BENGIO; HINTON, 2015).

2.2 Trabalhos Relacionados

Esta seção tem como objetivo situar o presente trabalho no contexto mais abrangente da pesquisa na área, examinando as contribuições relevantes e as estratégias empregadas por estudos prévios correlatos.

2.2.1 Detecção de Doença de Alzheimer a partir de Neuroimagens Utilizando Redes Neurais Convolucionais

O trabalho de Xavier (2018) tem como propósito desenvolver uma ferramenta computacional capaz de realizar a classificação de imagens médicas de pacientes com Alzheimer em diferentes estágios. Para alcançar esse objetivo, são utilizadas CNN na classificação de imagens de RM IRM, empregando o formato de arquivo NIFTI, que representa uma imagem tridimensional.

Na extração de características, foi adotada a utilização de uma rede Visual Geometry Group - 16 layers (VGG16). Isso se deve ao fato de que, dado o caráter minucioso das características diagnósticas para o Alzheimer, a arquitetura mais adequada consiste em filtros de convolução de baixa dimensão. Essa escolha visa identificar pontos que poderiam passar despercebidos em filtros de dimensões maiores.

Diversas métricas foram empregadas para avaliar a eficácia da classificação no projeto. A acurácia, que indica a precisão global do modelo, foi utilizada para medir a proporção de predições corretas em relação ao total de instâncias. A sensibilidade, ou recall, expressa a capacidade do modelo em identificar corretamente as instâncias positivas, enquanto a especificidade mensura sua habilidade em identificar corretamente as instâncias negativas, evitando Falsos positivos (FP). A precisão fornece uma medida de confiabilidade nas predições positivas, indicando a proporção de instâncias positivas corretamente identificadas em relação ao total previsto como positivo. Além disso, o F1-score, considerando tanto a precisão quanto a sensibilidade, oferece uma métrica balanceada que leva em conta FP e Falsos negativos (FN), sendo particularmente útil em casos de desequilíbrio entre as classes. Essas métricas, em conjunto, proporcionam uma avaliação abrangente do de-

sempenho do modelo em diferentes aspectos, permitindo uma análise mais completa da capacidade de classificação.

Como resultado da aplicação dos métodos de classificação binária em pacientes saudáveis e naqueles diagnosticados com Alzheimer, observaram-se aprimoramentos nos resultados. No entanto, é importante destacar que o trabalho indica que a classificação baseada apenas nas imagens apresenta limitações. Para um diagnóstico mais preciso, seria necessário incorporar outros dados clínicos, como resultados de exames de sangue e considerações sobre condições patológicas. Isso ressalta a importância de uma abordagem holística e integrada, combinando informações provenientes de diversas fontes para obter uma avaliação mais abrangente e precisa do estado de saúde dos pacientes.

2.2.2 Aprendizado de características para classificação de da doença de Alzheimer

Este trabalho de Ishibe (2022) discute a relevância da RM na detecção precoce da doença de Alzheimer e apresenta um projeto utilizando CNN para classificar cerebral. O objetivo é criar um modelo que possa classificar pacientes em diferentes estágios da doença. Esta lógica destaca a necessidade de métodos de detecção precoce mais eficazes, dadas as limitações do diagnóstico clínico tradicional, e enfatiza a capacidade das CNN de extrair padrões de imagens médicas. A questão de pesquisa busca identificar os algoritmos de classificação de imagens mais eficientes, com o objetivo específico de avaliar arquiteturas existentes e estabelecer uma linha de base para comparação de resultados. Em resumo, este projeto pretende contribuir para um diagnóstico mais preciso da DA, especialmente nas suas fases iniciais, aproveitando o potencial das CNN na análise de imagens de RM.

No âmbito da extração de características, destaca-se a transferência de aprendizado (transfer learning), uma técnica que aproveita arquiteturas ou modelos previamente treinados com grandes conjuntos de dados, como o ImageNet, para abordar tarefas em um domínio diferente. Para adaptar esses modelos preexistentes a uma nova tarefa, existem dois principais métodos: ajuste fino (fine-tuning) e extração de características (feature extraction). O ajuste fino envolve treinar o modelo pré-treinado no conjunto de dados específico da tarefa desejada, permitindo modificações apenas nos parâmetros das camadas mais profundas. Já no método de extração de características, a saída de uma camada específica do modelo pré-treinado é utilizada como entrada em uma nova arquitetura. Modelos amplamente empregados nesse contexto, encontrados frequentemente na literatura científica, incluem a Visual Geometry Group (VGG), GoogleNet e ResNet. Essas abordagens evidenciam a flexibilidade e eficácia da transferência de aprendizado na extração de características para tarefas específicas.

Nele ilustra a importância das métricas no desenvolvimento de Redes neurais profundas (MPL) e destaca a precisão como uma métrica chave em problemas de classificação binária,

representando a porcentagem de classificações corretas. A fórmula de precisão leva em consideração Verdadeiros positivos (TP), Verdadeiros negativos (TN), FP e FN. Além disso, o parágrafo menciona estender a precisão ao caso de classificação multiclasse, onde a métrica avalia o percentual de acertos em todas as classes. No contexto de análise de imagens médicas (por exemplo, RM no projeto proposto), a seleção e avaliação de técnicas de extração de características (por exemplo, CNN) desempenha um papel fundamental na busca de melhores resultados e na interpretação das informações contidas nas imagens. informação relacionada.

O estudo desenvolveu uma rede rasa com uma acurácia de 78,5% na classificação de IRM do cérebro, abrangendo diferentes estágios da doença de Alzheimer. O aumento do conjunto de dados e a aplicação de dropout contribuíram para a robustez do modelo, abordando desafios como desbalanceamento de classes e overfitting. A comparação com um baseline SVM treinado pela ResNet-50 revelou um desempenho superior da rede rasa. No entanto, a dificuldade persiste na classificação dos estágios mais leves da doença, refletindo a complexidade na identificação precoce do Alzheimer, mesmo em contextos clínicos. Esses resultados destacam avanços significativos e desafios enfrentados no desenvolvimento da abordagem proposta.

2.2.3 Utilização de Redes Convolucionais Para Classificação e Diagnóstico da Doença de Alzheimer

Neste trabalho de Silva et al. (2018), aborda-se a aplicação de CNN na classificação e diagnóstico do Mal de Alzheimer, uma doença neurodegenerativa prevalente que impacta a função mental e representa a principal causa de demência. Diante da relevância do diagnóstico precoce, os pesquisadores exploram o potencial das CNN, reconhecidas por sua capacidade de aprendizado em imagens, para a análise de ressonâncias magnéticas cerebrais. A pesquisa almeja contribuir para a eficácia dos métodos de diagnóstico por meio de experimentos minuciosos, evidenciando a estrutura da rede desenvolvida e seu papel crucial na análise de imagens médicas.

A extração de características neste trabalho é conduzida por camadas de convolução, que aplicam filtros ajustáveis automaticamente às imagens de entrada para identificar características cruciais à classificação. Quatro camadas de convolução são empregadas, seguidas por camadas de pooling que reduzem a dimensionalidade dos dados. A função de max pooling é utilizada, aplicando a função max em janelas 2x2 com valores determinados pela saída da camada convolucional. O treinamento da rede envolve ajustes contínuos nos filtros das camadas convolucionais ao longo de diversas épocas, e experimentos são realizados para determinar a quantidade ideal de épocas, variando de 5 a 50 em incrementos de 5. Essa abordagem dinâmica visa otimizar a extração de características, contribuindo para um desempenho aprimorado da rede convolucional na classificação de imagens médicas.

No projeto, diversas métricas são empregadas para avaliar a eficácia do classificador. A acurácia, a primeira métrica considerada, é calculada pela soma dos TP e FN, dividida pelo total de indivíduos no conjunto de teste. Essa métrica reflete com precisão a relação de acertos do classificador, indicando sua eficiência. Contudo, em situações de desbalanceamento de classes ou quando o classificador não oferece uma alta confiabilidade estatística, uma alta acurácia pode não refletir adequadamente a qualidade do modelo. Para abordar essa questão, são utilizadas as métricas de sensibilidade e especificidade, que analisam a proporção de TP e TN em relação ao conjunto de testes, respectivamente. Essas métricas são cruciais para diagnosticar a confiabilidade do modelo na predição de FP e TN, sendo este último especialmente relevante no contexto clínico. A combinação dessas métricas oferece uma avaliação abrangente do desempenho do classificador, considerando diferentes aspectos e contribuindo para uma análise mais completa e confiável.

O modelo, composto por quatro camadas convolucionais, alcançou uma acurácia máxima de 84,85%, com sensibilidade e especificidade de 0,8593 e 0,8314, respectivamente. Os resultados indicam eficiência, generalização e boa capacidade de acertos, superando alguns algoritmos clássicos. No entanto, a abordagem apresenta uma limitação relacionada ao tempo de treinamento da rede, sendo mais lento em comparação com algoritmos clássicos. Uma proposta para contornar essa limitação envolve treinar a base de dados em um servidor, enquanto o médico avaliador apenas envia as fatias correspondentes ao paciente a ser analisado.

2.2.4 Considerações finais

Com base nas pesquisas correlatas, buscou-se destacar quais CNN e técnicas de processamento de dados foram empregadas no diagnóstico da doença de Alzheimer. A Tabela 1 mostra as CNN e métodos de processamento de dados utilizadas.

Através da Tabela 1, observa-se que apenas o estudo de Ishibe (2022) faz uso de todas as redes convolucionais (CNN), empregando as arquiteturas VGG, GoogleNet e ResNet, além de realizar pré-processamento dos dados. Em contraste, a pesquisa de Silva et al. (2018) utiliza apenas a VGG16, limitando-se ao ajuste de parâmetros. Já o trabalho de (XAVIER, 2018), embora também empregue a VGG16, concentra-se exclusivamente no pré-processamento dos dados.

Dentro da ResNet, é possível ajustar vários parâmetros para buscar uma melhoria na eficiência do modelo. Isso inclui modificar o número de camadas, alterar o tamanho dos filtros convolucionais para capturar diferentes níveis de detalhe nas imagens, ajustar a taxa de aprendizado e definir o número de épocas. Essas modificações permitem adaptar a arquitetura da rede às características específicas do problema e otimizar seu desempenho na tarefa de diagnóstico de Alzheimer.

Em comparação com outros estudos, (SILVA et al., 2018) utilizou uma CNN, alcançando uma acurácia de 85%. Já (XAVIER, 2018) empregou a VGG16 com uma eficiência

Tabela 1 – Resumo Comparativo dos Trabalhos Relacionados

Trabalhos relacionados	CNN	VGG-16/VGG	GoogleNet	ResNet	Ajuste de Parâmetros	Pré-Processamento
(ISHIBE, 2022)	✓	✓	✓	✓	✗	✓
(SILVA et al., 2018)	✓	✗	✗	✗	✓	✗
(XAVIER, 2018)	✓	✓	✗	✗	✗	✓

variando entre 50% e 100%. É importante destacar que (ISHIBE, 2022) foi o único a utilizar a ResNet, embora outros estudos tenham obtido resultados interessantes.

Considerando essas informações, este trabalho explorará variações nos parâmetros da ResNet para observar sua performance aplicada a base de dados pública selecionada, para então avaliar para a aplicação de técnicas de pre-processamento (ISHIBE, 2022).

Experimentos e Análise dos Resultados

3.1 Metodologia

O capítulo “Experimentos e Análise dos Resultados” apresenta a aplicação prática do modelo ResNet para a classificação de imagens de ressonância magnética (MRI) associadas ao diagnóstico de Alzheimer, utilizando a base de dados “Alzheimer’s Dataset (4 class of Images)”. Inicialmente, são descritos os métodos empregados, incluindo variações nos parâmetros da ResNet, como o número de camadas, a taxa de aprendizado e o número de épocas de treinamento, com o objetivo de otimizar o desempenho do modelo. Em seguida, detalha-se o processo de implementação e as estratégias utilizadas para ajustar a rede e melhorar sua precisão.

O objetivo deste capítulo é avaliar a influência da variação dos parâmetros da CNN Rede Neural Residual (ResNet), conforme utilizada por (ISHIBE, 2022), empregando técnicas de pré-processamento propostas por (XAVIER, 2018). Espera-se que, ao abordar essas configurações, seja possível obter resultados distintos dos previamente alcançados, potencialmente resultando em uma maior eficiência no reconhecimento da DA ou ausência dela.

Os experimentos conduzidos são documentados com o objetivo de avaliar o impacto da variação dos parâmetros sobre a acurácia e a perda (*loss*) do modelo. Os resultados obtidos demonstram como diferentes combinações de configurações influenciam o desempenho na tarefa de classificação. Por fim, são discutidos os resultados alcançados em comparação com estudos anteriores, destacando os pontos de melhoria e as limitações observadas, proporcionando uma análise crítica da eficácia do modelo na detecção de Alzheimer em diferentes estágios da doença.

3.1.1 Seleção da base de dados

No que diz respeito aos métodos, será empregada a ResNet, utilizando a base de dados “Alzheimer’s Dataset (4 class of Images)”¹, que inclui quatro tipos diferentes de imagens de ressonância magnética (MRI), classificadas como:

- ☐ Demência leve;
- ☐ Demência moderada;
- ☐ Não demência;
- ☐ Demência muito leve.

3.1.2 Variação dos Parâmetros

Neste estudo, optou-se pela ResNet, inspirada no trabalho de (ISHIBE, 2022), uma vez que, em seu trabalho, foi utilizada a base de dados “Dataset Alzheimer”², na qual as imagens são classificadas nas mesmas categorias: Demência leve, Demência moderada, Não demência e Demência muito leve.

No trabalho de (ISHIBE, 2022), a ResNet demonstrou os melhores resultados de acurácia, atingindo 78,5%.

Caso necessário, serão adotadas técnicas de otimização de parâmetros ou pré-processamento de imagem, como feito nos estudos de (XAVIER, 2018) e (SILVA et al., 2018). Essa abordagem visa atingir resultados superiores ou inferiores, proporcionando entendimentos sobre a adequação desse método para o contexto em questão. Portanto, a sequência de métodos a seguir serão executadas:

1. Número de camadas: 20 camadas
2. Tamanho dos filtros convolucionais: 3x3
3. Taxa de aprendizado: 0,0001
4. Número de épocas: 200 épocas
5. Intervalo de variação dos parâmetros da rede:
 - ☐ Número de camadas: 18 a 50 camadas.
 - ☐ Tamanho dos filtros convolucionais: 3x3 a 5x5.
 - ☐ Taxa de aprendizado: 0,00001 a 0,001.
 - ☐ Número de épocas: 100 a 300 épocas.

¹ URL: <https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images/data>

² URL: <https://www.kaggle.com/datasets/yasserhessein/dataset-alzheimer>

6. Caso não seja verificado aumento considerável na eficiência será implementado a seguinte abordagem:

- ❑ implementação da técnica de pré-processamento usado por Xavier (2018): filtro de mediana;

3.1.3 Métodos para a Avaliação

Com o objetivo de extrair informações relacionadas à variação da eficiência na classificação dos padrões durante a variação dos parâmetros, será implementada uma codificação instrumentada para armazenar os valores associados à eficiência à medida que os parâmetros forem avaliados. As métricas utilizadas para essa análise serão a eficácia (*accuracy*) e a perda (*loss*) relacionadas aos testes.

3.2 Detalhes de Implementação

Nesta seção, será apresentado o processo de implementação da ResNet. O *code listing* 3.1 ilustra a definição dos parâmetros da CNN, começando pelo `batch_size`, que estabelece o número de amostras propagadas pela rede a cada iteração. O parâmetro `epochs` define o número de vezes que o modelo percorrerá todo o conjunto de dados durante o treinamento, permitindo que o modelo aprenda de forma mais eficaz a cada repetição.

O parâmetro `num_classes` especifica a quantidade de classes que o modelo deve ser capaz de classificar, refletindo a natureza do problema de classificação. A `learning_rate` determina a taxa de aprendizado, ou seja, o tamanho das atualizações feitas nos pesos da rede durante o treinamento, influenciando a velocidade e a estabilidade da convergência do modelo.

O parâmetro `depth` indica a profundidade da rede, representando o número de camadas presentes na arquitetura. A resolução das imagens de entrada é definida pelos parâmetros `img_rows` e `img_cols`, que, neste caso, possuem dimensões de 208x176 pixels. Por fim, `input_shape` especifica o formato de entrada para o modelo, indicando que as imagens possuem três canais (RGB) e uma resolução de 208x176 pixels.

Listing 3.1 – Definição dos parâmetros da CNN

```
batch_size = 32
epochs = 200
num_classes = 4
learning_rate = 0.0001
depth = 20

img_rows, img_cols = 208, 176
```

```
input_shape = (img_rows, img_cols, 3)
```

No *code listing* 3.2, temos uma função que implementa um filtro de mediana aplicado ao conjunto de imagens de Ressonância Magnética (MRI) com o objetivo de reduzir ruídos indesejados que pode interferir na qualidade e clareza das imagens.

A função `apply_median_filter` percorre todas as imagens do conjunto e aplica o filtro de mediana individualmente em cada uma delas. Esse processo consiste em analisar uma região ao redor de cada pixel, cujo tamanho é definido pelo parâmetro `kernel_size` (representando o tamanho da janela). Em seguida, a função calcula a mediana dos valores dos pixels dentro dessa região e substitui o valor do pixel central da janela pela mediana calculada.

Esse processo reduz a influência de pixels com valores extremos (pontos de ruído), preservando as características essenciais da imagem, como bordas e contornos. Dessa forma, o filtro de mediana ajuda a suavizar o ruído sem desfocar os detalhes importantes da imagem, o que é essencial para modelos de aprendizado de máquina, especialmente em tarefas que envolvem análise de imagens de MRI.

Listing 3.2 – Funcao do filtro de mediana no conjunto de iamgens

```
def apply_median_filter(images, kernel_size=3):
    filtered_images = images.copy()
    for i in range(images.shape[0]):
        filtered_images[i] = median_filter(images[i], size=(
            kernel_size, kernel_size, 1))
    return filtered_images
```

No *code listing* 3.3, temos uma função que implementa uma camada para a Rede Neural Residual (ResNet), configurada com um *kernel* de tamanho 5x5. Esta função, chamada `resnet_layer`, é projetada para construir um bloco convolucional, elemento essencial da arquitetura ResNet, que permite o aprendizado de características complexas em imagens.

A função aceita diversos parâmetros. O parâmetro `inputs` representa a entrada da camada, que pode ser a imagem original ou o resultado de uma camada anterior. O `num_filters` define o número de filtros convolucionais aplicados, determinando quantas características serão extraídas da entrada. O parâmetro `kernel_size` define o tamanho da janela de convolução (5x5 neste caso), que impacta a quantidade de informações que cada filtro pode processar de cada vez.

A função permite especificar `strides`, que determina o passo da janela ao aplicar a convolução, e `activation`, que define a função de ativação (neste caso, `relu`), essencial para introduzir não-linearidade e permitir que a rede aprenda relações complexas. O parâmetro `batch_normalization` controla a aplicação da normalização em lotes, uma

técnica usada para acelerar o treinamento e estabilizar a aprendizagem ao normalizar as saídas da camada convolucional.

Outro parâmetro relevante é o `weight_decay`, utilizado para regularizar os pesos da camada por meio da penalização L2, o que ajuda a evitar o *overfitting*. Além disso, o parâmetro `conv_first` determina a ordem das operações, configurando se a convolução será aplicada antes da normalização e ativação ou vice-versa.

Essa função retorna o *output* da camada, que pode ser usado em outras camadas para construir blocos residuais na ResNet. Assim, a função `resnet_layer` permite a criação de uma camada convolucional personalizada para otimizar o aprendizado de características de alto nível em imagens de Ressonância Magnética.

Listing 3.3 – Função para criar uma camada da ResNet com Kernel 5x5

```
def resnet_layer(inputs, num_filters=16, kernel_size=5,
                 strides=1, activation='relu', batch_normalization=True,
                 conv_first=True, weight_decay=1e-4):
    conv = Conv2D(num_filters, kernel_size=kernel_size,
                  strides=strides, padding='same',
                  kernel_initializer='he_normal',
                  kernel_regularizer=l2(weight_decay))

    x = inputs
    if conv_first:
        x = conv(x)
        if batch_normalization:
            x = BatchNormalization()(x)
        if activation is not None:
            x = Activation(activation)(x)
    else:
        if batch_normalization:
            x = BatchNormalization()(x)
        if activation is not None:
            x = Activation(activation)(x)
    x = conv(x)
    return x
```

No *code listing 3.4*, encontra-se a função `resnet_block`, que implementa um bloco residual. Este bloco é fundamental para facilitar o aprendizado em Deep Learning, permitindo a propagação de informações por meio de conexões de atalho. Essas conexões ajudam a mitigar problemas como o desaparecimento do gradiente, garantindo que características importantes das imagens sejam preservadas ao longo da rede.

A função `resnet_block` recebe quatro parâmetros: `inputs`, que representam os dados

de entrada do bloco residual; `num_filters`, que define a quantidade de filtros aplicados em cada camada convolucional da CNN do bloco; `num_blocks`, que determina a quantidade de camadas convolucionais a serem empilhadas, ampliando a capacidade de extração de padrões complexos; e `downsample`, que possibilita o ajuste da resolução espacial das imagens de entrada.

Dentro da função, um laço percorre a quantidade de camadas especificadas em `num_blocks`, aplicando duas operações convolucionais em cada iteração. A primeira operação convolucional é aplicada diretamente sobre a entrada `x`, com `stride` ajustado para 2, reduzindo a resolução espacial quando `downsample` é `True`. A segunda operação convolucional é aplicada na sequência, mantendo a estrutura do bloco e preservando a resolução para as próximas operações.

Caso `downsample` seja `True` e a camada seja a primeira do bloco (ou seja, `i == 0`), a entrada `x` passa por uma convolução `1x1` com `stride` 2. Essa convolução ajusta a dimensão da entrada para corresponder à saída de `y`, permitindo que a soma dos tensores ocorra corretamente. A soma residual, implementada por `add([x, y])`, combina as informações da entrada original e da saída convolucionada, garantindo que informações essenciais sejam preservadas ao longo das camadas, facilitando o fluxo de gradientes.

Finalmente, uma ativação `ReLU` é aplicada ao final do bloco, introduzindo uma não-linearidade essencial para a rede aprender padrões complexos e não-lineares. A estrutura do bloco residual é, portanto, crucial para a aprendizagem de características profundas em redes convolucionais, melhorando o desempenho em tarefas complexas e garantindo que redes mais profundas sejam eficazes.

Listing 3.4 – Função para criar um bloco residual

```
def resnet_block(inputs, num_filters=16, num_blocks=3,
                downsample=False):
    x = inputs
    for i in range(num_blocks):
        strides = 2 if i == 0 and downsample else 1
        y = resnet_layer(x, num_filters=num_filters, strides=
                        strides)
        y = resnet_layer(y, num_filters=num_filters,
                        activation=None)
        if i == 0 and downsample:
            x = resnet_layer(x, num_filters=num_filters,
                            kernel_size=1, strides=strides, activation=
                            None, batch_normalization=False)
        x = add([x, y])
        x = Activation('relu')(x)
    return x
```

No *code listing* 3.5, encontra-se a função `resnet_model`, que implementa um modelo ResNet com um *kernel* convolucional de tamanho 5x5. Essa função permite a configuração do modelo utilizando três parâmetros principais: `input_shape`, que define o formato das entradas do modelo; `depth`, que determina a profundidade da rede; e `num_classes`, que representa a quantidade de classes-alvo que o modelo deve classificar.

A função começa com a definição do número inicial de filtros, `num_filters`, configurado como 16, e calcula o número de blocos residuais, `num_res_blocks`, com base na profundidade desejada. A camada de entrada é criada com o uso de `Input`, utilizando a forma indicada em `input_shape`. Em seguida, aplica-se uma primeira camada convolucional, por meio da função `resnet_layer`, que utiliza um `kernel_size` de 5x5 para extrair características iniciais da imagem.

A estrutura principal do modelo é formada por dois laços aninhados, que controlam a construção dos blocos residuais em três “pilhas” (`stack`). O primeiro laço percorre essas pilhas, enquanto o segundo laço constrói o número definido de blocos residuais em cada uma. O parâmetro `downsample` é configurado como `True` somente quando a resolução espacial precisa ser reduzida, o que ocorre ao iniciar uma nova pilha, permitindo que a rede processe características de diferentes escalas ao longo das camadas.

Após a construção de todos os blocos residuais, aplica-se uma camada de `AveragePooling2D`, utilizada para reduzir a dimensionalidade. Em seguida, uma camada de `Flatten` transforma as características extraídas em uma única dimensão, facilitando a entrada nas camadas densas subsequentes. Após essas duas etapas, uma camada de `Dropout` com taxa de 0,5 é inserida para minimizar o risco de *overfitting* e melhorar a capacidade de generalização do modelo. Por fim, uma camada densa de saída, com ativação `softmax`, é aplicada para gerar as probabilidades de cada classe, completando assim o processo de classificação e retornando o modelo ResNet.

Listing 3.5 – Funcao para cirar o modelo ResNet com Kernel 5x5

```
def resnet_model(input_shape, depth, num_classes=4):
    num_filters = 16
    num_res_blocks = (depth - 2) // 6
    inputs = Input(shape=input_shape)
    x = resnet_layer(inputs, num_filters=num_filters,
                     kernel_size=5) # Kernel 5x5
    for stack in range(3):
        for res_block in range(num_res_blocks):
            downsample = True if stack > 0 and res_block == 0
                           else False
            x = resnet_block(x, num_filters=num_filters,
                             downsample=downsample)
```

```
num_filters *= 2
x = AveragePooling2D(pool_size=(2, 2))(x)
x = Flatten()(x)
x = Dropout(0.5)(x)
outputs = Dense(num_classes, activation='softmax',
                 kernel_initializer='he_normal')(x)
model = Model(inputs=inputs, outputs=outputs)
return model
```

No *code listing* 3.6, é mostrada a etapa de compilação do modelo ResNet, onde são configurados os parâmetros de treinamento. Primeiramente, a função `resnet_model` utiliza os parâmetros `input_shape`, `depth` e `num_classes`, que especificam o formato dos dados de entrada, a profundidade da rede e o número de classes para classificação.

A função de perda escolhida é a `categorical_crossentropy`. Essa função mede a diferença entre as probabilidades previstas pelo modelo e as classes verdadeiras, ajustando os pesos para minimizar essa discrepância.

O otimizador `Adam` é configurado com uma taxa de aprendizado definida pela variável `learning_rate`. Este otimizador combina as vantagens dos algoritmos `RMSprop` e `momentum`, proporcionando atualizações de pesos mais rápidas e eficientes.

Por fim, a métrica `accuracy` é utilizada para monitorar o desempenho do modelo durante o treinamento, indicando a proporção de previsões corretas e facilitando o acompanhamento da evolução do modelo enquanto os pesos são ajustados.

Listing 3.6 – Compilação do modelo

```
model = resnet_model(input_shape=input_shape, depth=depth,
                     num_classes=num_classes)
model.compile(loss='categorical_crossentropy',
              optimizer=Adam(learning_rate=learning_rate),
              metrics=['accuracy'])
```

No *code listing* 3.7, são mostradas as configurações dos `callbacks`. Primeiramente, o diretório onde o modelo será salvo é definido utilizando a função `os.path.join` para combinar o diretório atual obtido pela função `os.getcwd()`.

Em seguida, é verificado se o diretório `save_dir` existe; caso não exista, ele é criado utilizando da função `os.makedirs(save_dir)`. O caminho completo para o arquivo do modelo é construído utilizando `os.path.join` novamente.

O `checkpoint` é configurado por meio da classe `ModelCheckpoint`, que salva o modelo durante o treinamento. O parâmetro `monitor` é definido como `val_accuracy`, fazendo com que o `checkpoint` salve apenas os melhores modelos com base na acurácia de validação. O parâmetro `verbose` é configurado como 1 para exibir mensagens informativas durante o salvamento.

A função `LearningRateScheduler` é utilizada para ajustar a taxa de aprendizado ao longo das épocas. A taxa de aprendizado é reduzida pela metade a cada 50 épocas, conforme especificado pela função anônima `lambda`.

O callback `ReduceLROnPlateau` também é configurado, permitindo que a taxa de aprendizado seja reduzida em um fator de 0.5 caso não haja melhoria na acurácia de validação por 10 épocas consecutivas.

Por fim, todos os callbacks são armazenados em uma lista chamada `callbacks`, que será passada para o método de treinamento do modelo.

Listing 3.7 – Configurações de salvamento de modelo e callbacks

```
save_dir = os.path.join(os.getcwd(), 'saved_models')
model_name = 'mri_resnet_model.keras'
if not os.path.isdir(save_dir):
    os.makedirs(save_dir)
filepath = os.path.join(save_dir, model_name)

checkpoint = ModelCheckpoint(filepath=filepath, monitor='
    val_accuracy', verbose=1, save_best_only=True)
lr_scheduler = LearningRateScheduler(lambda epoch:
    learning_rate * (0.5 ** (epoch // 50)))
lr_reducer = ReduceLROnPlateau(factor=0.5, patience=10,
    verbose=1)
callbacks = [checkpoint, lr_reducer, lr_scheduler]
```

No *code listing* 3.8, são apresentados os geradores de dados para o treinamento e validação do modelo. Inicialmente, são criados dois objetos da classe `ImageDataGenerator`, sendo `train_datagen` para os dados de treinamento e `test_datagen` para os dados de teste.

O método `flow_from_directory` é utilizado para carregar as imagens a partir dos diretórios. As imagens utilizadas para o gerador de treinamento `train_generator` são carregadas a partir do diretório `train_dir`, com as dimensões previamente ajustadas para `img_rows = 208` e `img_cols = 176`. O parâmetro `batch_size` define o número de amostras que serão processadas em cada iteração, enquanto `class_mode = 'categorical'` especifica que a saída será em formato categórico.

De forma semelhante, o gerador de validação `validation_generator` é configurado utilizando o diretório `validation_dir`, mantendo os mesmos parâmetros de ajuste de tamanho, tamanho do lote e modo de classe.

Listing 3.8 – Geradores de dados

```
train_datagen = ImageDataGenerator()
test_datagen = ImageDataGenerator()
```

```
train_generator = train_datagen.flow_from_directory(  
    train_dir,  
    target_size=(img_rows, img_cols),  
    batch_size=batch_size,  
    class_mode='categorical')  
  
validation_generator = test_datagen.flow_from_directory(  
    validation_dir,  
    target_size=(img_rows, img_cols),  
    batch_size=batch_size,  
    class_mode='categorical')
```

No *code listing* 3.9, é apresentado o processo de treinamento do modelo, iniciado pelo método `fit` e os parâmetros definidos pela classe `model`.

Inicialmente, foram definidos os parâmetros `train_generator`, que fornece os dados de treinamento, e `validation_generator`, que fornece os dados de validação. O parâmetro `steps_per_epoch` é calculado utilizando o número total de amostras de treinamento dividido pelo `batch_size`, indicando quantas iterações de treinamento serão realizadas em cada época.

De maneira semelhante, `validation_steps` é configurado com o cálculo do número total de amostras de validação dividido pelo `batch_size`, especificando quantas iterações de validação serão realizadas por época.

O treinamento é executado por um número definido de épocas, determinado pelo parâmetro `epochs`, que foi definido no *code listing* 3.1. O parâmetro `verbose` é definido como 1 para exibir informações sobre o progresso do treinamento. Além disso, a lista de `callbacks` é passada como um parâmetro, permitindo o uso de funcionalidades adicionais durante o treinamento, incluindo o salvamento do modelo e a adaptação da taxa de aprendizado.

Listing 3.9 – Treinamento do modelo

```
model.fit(train_generator,  
          steps_per_epoch=train_generator.samples //  
            batch_size,  
          validation_data=validation_generator,  
          validation_steps=validation_generator.samples //  
            batch_size,  
          epochs=epochs, verbose=1,  
          callbacks=callbacks)
```

3.3 Resultados Alcançados

Conforme descrito na Seção 3.1.1, são apresentados os resultados iniciais da verificação do impacto da variação dos parâmetros na acurácia da rede, conforme mostrado na Tabela 2. Esses resultados refletem como diferentes combinações de camadas, épocas de treinamento e taxas de aprendizado afetam o desempenho da rede convolucional ao realizar a classificação. A análise desses parâmetros é essencial para ajustar o modelo e maximizar sua precisão, permitindo identificar as configurações mais adequadas para obter melhores resultados na tarefa de classificação.

Tabela 2 – Resultados da Variação dos Parâmetros para Filtros Convolucionais 3x3

Parâmetros Variados								
Camadas			Épocas			Taxa de Aprendizado		
Testes / Valor	Acurácia	Perda	Testes / Valor	Acurácia	Perda	Teste/Valor	Acurácia	Perda
Teste 01: 20	44.99%	12.27	Teste 06: 100	10.00%	16.86	Teste 11: 0.1	30.00%	1.56
Teste 02: 29	30.00%	18.03	Teste 07: 150	25.00%	12.44	Teste 12: 0.01	30.00%	2.03
Teste 03: 34	25.00%	13.94	Teste 08: 200	44.99%	18.03	Teste 13: 0.001	25.00%	20.01
Teste 04: 38	25.00%	14.67	Teste 09: 250	20.00%	19.06	Teste 14: 0.0001	44.99%	18.03
Teste 05: 47	15.00%	20.92	Teste 10: 300	40.00%	16.32	Teste 15: 0.00001	20.00%	11.08

Pode-se observar que os **Testes 01, 08 e 14** atingiram uma acurácia de 44,99%, com as seguintes configurações: 20 camadas neurais, 200 épocas de treinamento e taxa de aprendizado de 0,0001. O **Teste 10** obteve uma acurácia de 40,00%, utilizando 300 épocas de treinamento, 20 camadas neurais e taxa de aprendizado de 0,0001. Já os **Testes 11 e 12** apresentaram uma acurácia de 30%, com configurações de taxa de aprendizado de 0,0001 e 0,00001, 20 camadas neurais e 200 épocas de treinamento.

A variação dos parâmetros com o uso de filtros convolucionais 3x3 não foi suficiente para alcançar a métrica de 78,5% obtida por (ISHIBE, 2022). Contudo, os melhores percentuais de acurácia para cada parâmetro podem ser observados pelos itens destacados em negrito na Tabela 2.

Tendo em vista a baixa acurácia, foram realizados novos testes com os parâmetros na ResNet 5x5, utilizando-se os mesmos valores de camadas, épocas e taxa de aprendizado aplicados nos testes da ResNet 3x3. Dessa forma, obtiveram-se os dados apresentados na Tabela 3.

Pode-se observar que os **Testes 01, 03 e 04** obtiveram a mesma acurácia de 25%. Contudo, para definir o melhor valor de camadas, utilizou-se a menor perda como critério de desempate, com o melhor resultado sendo alcançado com 20 camadas, 200 épocas e uma taxa de aprendizado de 0,0001.

O **Teste 06** apresentou o melhor resultado na variação de épocas, atingindo uma acurácia de 44,99% ao utilizar 20 camadas, 100 épocas e uma taxa de aprendizado de 0,0001.

No **Teste 12**, obteve-se uma acurácia de 34,99%, utilizando os parâmetros de 20 camadas, 100 épocas e uma taxa de aprendizado de 0,01. O **Teste 14** apresentou a

mesma acurácia do **Teste 06**, pois os parâmetros utilizados eram os mesmos.

Entretanto, o melhor resultado foi obtido no **Teste 11**, com uma acurácia de 50%, utilizando 20 camadas, 100 épocas e uma taxa de aprendizado de 0,1. Este resultado representou uma melhora de 5,01% em relação ao melhor teste obtido na ResNet 3x3.

Tabela 3 – Resultados da Variação dos Parâmetros para Filtros Convolucionais 5x5

Parâmetros Variados					
Camadas		Épocas		Taxa de Aprendizado	
Testes / Valor Parâmetro	Acurácia	Testes / Valor Parâmetro	Acurácia	Teste/Valor Parâmetro	Acurácia
Teste 01: 20	25.00%	Teste 06: 100	44.99%	Teste 11: 0.1	50.00%
Teste 02: 29	15.00%	Teste 07: 150	30.00%	Teste 12: 0.01	34.99%
Teste 03: 34	25.00%	Teste 08: 200	25.00%	Teste 13: 0.001	25.00%
Teste 04: 38	25.00%	Teste 09: 250	30.00%	Teste 14: 0.0001	44.99%
Teste 05: 47	20.00%	Teste 10: 300	30.00%	Teste 15: 0.00001	15.00%

Após a realização dos testes com as arquiteturas ResNet 3x3 e 5x5, a melhor acurácia encontrada foi de 50% utilizando a ResNet 5x5. A variação dos parâmetros não foi suficiente para alcançar a métrica de 78,5% obtida por (ISHIBE, 2022). Conforme mencionado anteriormente, foram realizados testes adicionais utilizando a técnica de pré-processamento de (XAVIER, 2018), implementando o filtro de mediana.

O teste com o filtro de mediana foi aplicado apenas ao melhor resultado encontrado, com parâmetros de 20 camadas, 100 épocas e uma taxa de aprendizado de 0,1. A acurácia do teste com o filtro de mediana permaneceu em 50%, porém a perda (*loss*) do teste foi reduzida, passando de 6,33 para 4,54.

3.4 Discussão e Avaliação dos Resultados

Os resultados dos experimentos com as arquiteturas ResNet 3x3 e 5x5 revelaram tanto avanços quanto limitações relevantes no processo de classificação de imagens de RM para o diagnóstico de Alzheimer. A melhor acurácia alcançada, de 50%, foi obtida com a ResNet 5x5, utilizando 20 camadas, 100 épocas e uma taxa de aprendizado de 0.1, superando todas as configurações testadas na ResNet 3x3, cuja melhor acurácia foi de 44,99%.

Para contextualizar esses resultados, é importante considerar que o dataset utilizado possui quatro classes de classificação — Não Demência, Demência Muito Leve, Demência Leve e Demência Moderada —, o que implica que um classificador aleatório obteria, em média, 25% de acurácia. Nesse sentido, o resultado de 50% da ResNet 5x5 indica que o modelo aprendeu padrões relevantes nas imagens, superando em 25 pontos percentuais o desempenho esperado por acaso. No entanto, esse valor ainda está distante de uma aplicação clínica confiável, evidenciando a necessidade de aprimoramentos na abordagem adotada.

A análise comparativa entre as duas arquiteturas sugere que os filtros de maior dimensão (5x5) foram mais adequados para capturar as características estruturais presentes nas imagens de RM, possivelmente por abrangerem regiões mais amplas da imagem a cada operação de convolução, o que pode favorecer a identificação de padrões morfológicos de maior escala, como a atrofia cerebral característica da DA (MAIA et al., 2019). Em contrapartida, os filtros 3x3, embora sejam eficazes na detecção de detalhes finos, podem ter sido insuficientes para capturar as variações estruturais mais amplas presentes nas imagens analisadas.

A exploração de diferentes combinações de camadas, épocas de treinamento e taxas de aprendizado permitiu identificar tendências importantes sobre o comportamento da rede. Em relação ao número de camadas, observou-se que redes mais profundas não resultaram necessariamente em melhor desempenho. Na ResNet 3x3, por exemplo, o aumento de 20 para 47 camadas reduziu a acurácia de 44,99% para 15%, sugerindo que, para o volume de dados disponível neste estudo, arquiteturas excessivamente profundas tendem a apresentar dificuldades de convergência ou sobreajuste aos dados de treinamento (SILVA et al., 2018). Esse comportamento é consistente com a literatura, que aponta que a profundidade ótima de uma rede está diretamente relacionada à quantidade e à diversidade dos dados utilizados no treinamento (ISHIBE, 2022).

Em relação à taxa de aprendizado, os resultados indicaram que valores muito baixos, como 0,00001, prejudicaram o desempenho do modelo em ambas as arquiteturas, resultando em acurácias de 20% e 15% para as ResNets 3x3 e 5x5, respectivamente. Esse comportamento sugere que taxas de aprendizado muito reduzidas podem ter levado o modelo a uma convergência prematura em mínimos locais subótimos, comprometendo sua capacidade de generalização. Por outro lado, a taxa de 0,1, que produziu o melhor resultado geral na ResNet 5x5, indica que valores mais elevados podem ter favorecido uma exploração mais ampla do espaço de parâmetros durante o treinamento, contribuindo para um melhor ajuste do modelo (SILVA et al., 2018).

Quanto à variação do número de épocas, observou-se que nem o aumento nem a redução isolados foram suficientes para melhorar consistentemente a acurácia. Na ResNet 3x3, o melhor resultado com variação de épocas foi obtido com 200 épocas (44,99%), enquanto 100 épocas resultaram na pior acurácia desta arquitetura (10%). Na ResNet 5x5, o comportamento foi inverso, com 100 épocas produzindo o melhor resultado geral (44,99% na variação de épocas). Isso reforça que os parâmetros da rede interagem entre si de forma não trivial, e que a otimização isolada de cada um deles apresenta limitações inerentes (ISHIBE, 2022).

A aplicação do filtro de mediana, técnica de pré-processamento proposta por Xavier (2018), foi testada sobre a melhor configuração encontrada (ResNet 5x5 com 20 camadas, 100 épocas e taxa de aprendizado de 0,1). Embora a acurácia tenha permanecido em 50%, observou-se uma redução na perda do modelo, que passou de 6,33 para 4,54. Essa

redução indica que o filtro contribuiu para suavizar ruídos presentes nas imagens, tornando o processo de aprendizado mais estável. O fato de a acurácia não ter aumentado, contudo, sugere que o ruído não era o principal fator limitante do desempenho do modelo, e que outros aspectos, como o volume de dados e a estratégia de treinamento, exercem influência mais determinante sobre a capacidade de classificação da rede.

Em comparação com os trabalhos relacionados, o resultado obtido ficou abaixo das acurácias reportadas por Silva et al. (2018) (84,85%) e Ishibe (2022) (78,5%). É importante ressaltar, contudo, que essas comparações possuem limitações metodológicas relevantes: os estudos citados utilizaram datasets distintos do empregado neste trabalho, com diferentes distribuições de classes, volumes de dados e protocolos de aquisição de imagens. Dessa forma, diferenças nos resultados não podem ser atribuídas exclusivamente às escolhas arquiteturais, devendo ser interpretadas com cautela (ISHIBE, 2022; SILVA et al., 2018).

As principais limitações identificadas neste estudo que podem ter contribuído para o desempenho abaixo do esperado são:

- ❑ **Ausência de técnicas de aumento de dados (*data augmentation*):** o treinamento foi realizado sem aplicação de transformações artificiais nas imagens, como rotações, espelhamentos ou variações de brilho. Essas técnicas são amplamente utilizadas na literatura para ampliar artificialmente o volume de dados disponíveis e melhorar a capacidade de generalização do modelo (ISHIBE, 2022).
- ❑ **Métricas de avaliação restritas:** a análise foi conduzida com base exclusivamente na acurácia e na perda. A inclusão de métricas complementares, como sensibilidade, especificidade e F1-score por classe, permitiria uma avaliação mais detalhada do comportamento do modelo frente ao desbalanceamento entre as classes do dataset (XAVIER, 2018).

Apesar das limitações apontadas, os resultados obtidos cumprem o objetivo central deste trabalho, que foi analisar a influência dos parâmetros estruturais da ResNet antes da aplicação de técnicas mais sofisticadas de otimização. A identificação da configuração com melhor desempenho — ResNet 5x5 com 20 camadas, 100 épocas e taxa de aprendizado de 0,1 — estabelece um ponto de partida concreto para investigações futuras, reduzindo o espaço de busca de hiperparâmetros e direcionando esforços para as abordagens com maior potencial de melhoria.

Conclusão

4.1 Conclusão

Este trabalho teve como objetivo analisar a influência dos principais parâmetros estruturais da arquitetura ResNet no diagnóstico da doença de Alzheimer por meio de imagens de Ressonância Magnética, utilizando uma base de dados pública. Para isso, foram conduzidos experimentos sistemáticos com as variantes ResNet 3x3 e 5x5, explorando diferentes combinações de número de camadas, épocas de treinamento e taxas de aprendizado, além da aplicação do filtro de mediana como técnica de pré-processamento.

Em relação aos objetivos específicos propostos, todos foram cumpridos ao longo do desenvolvimento do trabalho. A base de dados pública “Alzheimer’s Dataset (4 class of Images)” foi selecionada e utilizada nos experimentos. Os principais parâmetros envolvidos no desempenho da rede foram estudados e variados sistematicamente, permitindo aferir o impacto de cada configuração sobre a acurácia e a perda do modelo. Por fim, a configuração com melhor desempenho foi identificada — ResNet 5x5 com 20 camadas, 100 épocas e taxa de aprendizado de 0.1 — e submetida à aplicação do filtro de mediana, encerrando o ciclo metodológico proposto.

Retomando as hipóteses levantadas no Capítulo 1, foi possível identificar que a ResNet 5x5 se mostrou o método mais efetivo entre as arquiteturas testadas, atingindo uma acurácia de 50% frente aos 44,99% obtidos pela ResNet 3x3. Esse resultado indica que filtros convolucionais de maior dimensão foram mais adequados para capturar os padrões estruturais presentes nas imagens de RM, possivelmente por abrangerem regiões mais amplas a cada operação de convolução, favorecendo a identificação de alterações morfológicas características da DA (MAIA et al., 2019). Em relação às melhorias que poderiam aumentar o percentual de predição, os experimentos evidenciaram que a variação isolada de parâmetros e a aplicação do filtro de mediana não foram suficientes para superar a barreira dos 50%, apontando que abordagens como *data augmentation*, validação cruzada e exploração conjunta de parâmetros por meio de *grid search* representam os caminhos mais promissores para ganhos expressivos de desempenho (ISHIBE, 2022).

É importante destacar que, embora a acurácia obtida esteja abaixo dos resultados reportados por Ishibe (2022) (78,5%) e Silva et al. (2018) (84,85%), essa comparação deve ser interpretada com cautela, uma vez que os estudos citados utilizaram datasets distintos, com diferentes volumes de dados, distribuições de classes e protocolos de aquisição de imagens. Nesse contexto, a acurácia de 50% obtida representa um resultado superior ao esperado por um classificador aleatório — que atingiria 25% em um problema de quatro classes —, indicando que o modelo foi capaz de aprender padrões relevantes nas imagens, ainda que com capacidade de generalização limitada.

A principal contribuição deste trabalho para a comunidade científica está no mapeamento do comportamento dos parâmetros estruturais da ResNet aplicada a um dataset público de imagens de RM para diagnóstico de Alzheimer, antes da adoção de técnicas mais sofisticadas de otimização. Os resultados gerados estabelecem um ponto de partida concreto para pesquisas futuras, reduzindo o espaço de busca de hiperparâmetros e evidenciando que a configuração com filtros 5x5, 20 camadas e taxa de aprendizado de 0.1 representa a base mais promissora para a continuidade das investigações (SILVEIRA, 2019).

4.2 Trabalhos Futuros

Diante dos resultados e limitações identificados ao longo deste trabalho, futuras pesquisas podem explorar as seguintes direções, diretamente motivadas pelos achados aqui apresentados:

- ❑ **Exploração conjunta de parâmetros:** a adoção de técnicas como *grid search* ou *random search* permitiria avaliar combinações entre número de camadas, taxa de aprendizado e épocas de forma simultânea, potencialmente identificando configurações mais eficientes do que as encontradas nos experimentos realizados neste trabalho (ISHIBE, 2022);
- ❑ **Aplicação de *data augmentation*:** a ausência de técnicas de aumento de dados foi identificada como uma das principais limitações deste trabalho. A aplicação de transformações como rotações, espelhamentos e variações de contraste nas imagens de RM poderia ampliar artificialmente o volume de dados disponíveis para treinamento, contribuindo para uma melhor generalização do modelo (ISHIBE, 2022);
- ❑ **Investigação de filtros convolucionais maiores:** considerando que a ResNet 5x5 superou a versão 3x3, seria relevante avaliar o comportamento de filtros de dimensões superiores, como 7x7, verificando se a tendência de melhoria se mantém ou se existe um ponto de saturação para este dataset específico;

- ❑ **Inclusão de métricas complementares:** a adoção de métricas como sensibilidade, especificidade, F1-score e matriz de confusão por classe permitiria uma avaliação mais detalhada do comportamento do modelo frente às quatro categorias do dataset, especialmente nos estágios mais leves da doença, onde a classificação tende a ser mais desafiadora (XAVIER, 2018);
- ❑ **Aplicação de técnicas avançadas de pré-processamento:** embora o filtro de mediana tenha contribuído para a redução da perda, técnicas como normalização de histograma, correção de campo de polarização (*bias field correction*) e segmentação cerebral automática poderiam fornecer imagens de melhor qualidade para o treinamento da rede (XAVIER, 2018).

A continuidade dessas investigações, partindo da configuração ótima identificada neste trabalho, poderá contribuir de forma significativa para o desenvolvimento de modelos mais eficientes na detecção do Alzheimer, auxiliando no diagnóstico precoce da doença e na tomada de decisões clínicas mais precisas.

Referências

- ASSOCIATION, A. 2019 alzheimer's disease facts and figures. **Alzheimer's & dementia**, Wiley Online Library, v. 15, n. 3, p. 321–387, 2019. Citado na página 15.
- ASSOCIATION, A. et al. 2016 alzheimer's disease facts and figures. **Alzheimer's & Dementia**, Elsevier, v. 12, n. 4, p. 459–509, 2016. Citado na página 13.
- DELABY, C.; TEUNISSEN, C. E. Blood-based biomarkers for alzheimer's disease: Challenges and perspectives. **Frontiers in Aging Neuroscience**, Frontiers Media SA, v. 15, 2023. Citado na página 12.
- FOROUGHIPOUR, S. et al. Deep learning approaches for alzheimer's disease diagnosis using mri images. **Healthcare**, v. 11, n. 3, p. 421, 2023. Citado na página 16.
- FRISONI, G. B. et al. The clinical use of structural mri in alzheimer disease. **Nature Reviews Neurology**, v. 18, n. 4, p. 197–213, 2022. Citado 2 vezes nas páginas 15 e 16.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [S.l.]: MIT Press, 2016. Citado na página 17.
- HAMPEL, H. et al. Revolution of alzheimer precision neurology: Passageway of systems biology and neurophysiology. **Journal of Prevention of Alzheimer's Disease**, v. 10, n. 2, p. 210–233, 2023. Citado na página 12.
- ISHIBE, E. I. Aprendizado de características para classificação de imagens de ressonância magnética da doença de alzheimer. 2022. Citado 11 vezes nas páginas 19, 21, 22, 23, 24, 33, 34, 35, 36, 37 e 38.
- JACK, C. R. et al. Updated criteria for diagnosis and staging of alzheimer's disease. **Alzheimer's & Dementia**, v. 19, n. 5, p. 1234–1248, 2023. Citado na página 16.
- KALES, H. C.; GITLIN, L. N.; LYKETSOS, C. G. Assessment and management of behavioral and psychological symptoms of dementia. **Bmj**, British Medical Journal Publishing Group, v. 350, 2015. Citado na página 13.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. Imagenet classification with deep convolutional neural networks. **Advances in Neural Information Processing Systems**, v. 25, 2012. Citado na página 17.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, v. 521, n. 7553, p. 436–444, 2015. Citado 2 vezes nas páginas 17 e 18.

- MAIA, M. C. et al. Inteligência artificial para o apoio ao diagnóstico da doença de alzheimer utilizando imagens de ressonância magnética. **Anais do III Simpósio de Inovação em Engenharia Biomédica-SABIO**, v. 2019, p. 47, 2019. Citado 3 vezes nas páginas 12, 35 e 37.
- NAIR, V.; HINTON, G. Rectified linear units improve restricted boltzmann machines. In: **Proceedings of the 27th International Conference on Machine Learning**. [S.l.: s.n.], 2010. p. 807–814. Citado na página 17.
- NITRINI, R. et al. Diagnóstico de doença de alzheimer no brasil: avaliação cognitiva e funcional. recomendações do departamento científico de neurologia cognitiva e do envelhecimento da academia brasileira de neurologia. **Arquivos de Neuro-psiquiatria, SciELO Brasil**, v. 63, p. 720–727, 2005. Citado na página 13.
- ORGANIZATION, W. H. et al. First who ministerial conference on global action against dementia: meeting report, who headquarters, geneva, switzerland, 16-17 march 2015. World Health Organization, 2015. Citado na página 12.
- RODRIGUES, P. M. **Diagnóstico da doença de alzheimer com base no electroencefalograma**. Tese (Doutorado) — Instituto Politécnico de Bragança, Escola Superior de Tecnologia e Gestão, 2011. Citado na página 12.
- RUDROFF, T.; RAINIO, O.; KLÉN, R. Artificial intelligence for early prediction of alzheimer’s disease using neuroimaging biomarkers. **Diagnostics**, v. 14, n. 2, p. 155, 2024. Citado na página 16.
- SALEHI, A. W. et al. A deep learning model for classification of alzheimer disease and mild cognitive impairment. **Biomedical Signal Processing and Control**, v. 57, p. 101810, 2020. Citado na página 16.
- SILVA, I. R. et al. Utilização de redes convolucionais para classificação e diagnóstico da doença de alzheimer. **II Simpósio de Inovação em Engenharia Biomédica**, p. 73–76, 2018. Citado 7 vezes nas páginas 20, 21, 22, 24, 35, 36 e 38.
- SILVEIRA, R. C. Análise de arquiteturas de redes neurais convolucionais para classificação de sinais de trânsito. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais Campus Formiga, 2019. Citado 2 vezes nas páginas 14 e 38.
- VARESI, A. et al. Blood-based biomarkers for alzheimer’s disease diagnosis and progression: An overview. **Cells**, MDPI, v. 11, n. 8, p. 1367, 2022. Citado na página 12.
- World Health Organization. **Global Status Report on the Public Health Response to Dementia**. Geneva: [s.n.], 2023. Citado 2 vezes nas páginas 13 e 15.
- XAVIER, M. F. Detecção de doença de alzheimer a partir de neuroimagens utilizando redes neurais convolucionais. 2018. Citado 10 vezes nas páginas 18, 21, 22, 23, 24, 25, 34, 35, 36 e 39.