

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Aprendizado de características para classificação de imagens de ressonância magnética da doença de Alzheimer

Eder Issao Ishibe

Trabalho de Conclusão de Curso
MBA em Inteligência Artificial e Big Data

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Aprendizado de características para
classificação de imagens de
ressonância magnética da doença
de Alzheimer

Eder Issao Ishibe

USP - São Carlos
2022

Eder Issao Ishibe

Título do trabalho: Aprendizado de características para classificação de imagens de ressonância magnética da doença de Alzheimer

Trabalho de conclusão de curso apresentado ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Fernando Pereira dos Santos

USP - São Carlos

2022

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

I79a Ishibe, Eder Issao
Aprendizado de características para classificação
de imagens de ressonância magnética da doença de
Alzheimer / Eder Issao Ishibe; orientador Fernando
Pereira dos Santos. -- São Carlos, 2022.
67 p.

Trabalho de conclusão de curso (MBA em
Inteligência Artificial e Big Data) -- Instituto de
Ciências Matemáticas e de Computação, Universidade
de São Paulo, 2022.

1. Redes Neurais. 2. Doença de Alzheimer. 3.
Ressonância magnética. I. dos Santos, Fernando
Pereira, orient. II. Título.

DEDICATÓRIA

*A minha avó que sofre dessa doença
terrível que é o Alzheimer*

AGRADECIMENTOS

Aos meus pais e irmãos, por todo o apoio que sempre tive em toda minha vida.

À minha amiga, companheira e esposa Tatiana, pela compreensão, força e carinho.

Aos meus amigos Daniel Moreira Cestari e Tales Roberto de Souza Santini, pelas ideias e sugestões sobre o trabalho, além das longas e boas conversas.

Ao meu amigo Prof. Dr. João Navarro Soares Júnior, pelos ensinamentos sobre pesquisa científica e redação, pelos quais sempre serei grato.

Ao meu orientador Prof. Dr. Fernando Pereira dos Santos, pela sua disponibilidade, competência e paciência que me auxiliou e guiou durante todo o desenvolvimento deste trabalho

RESUMO

ISHIBE, E. I. Aprendizado de características para classificação de imagens de ressonância magnética da doença de Alzheimer. 2022. 67 f. Trabalho de conclusão de curso (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2022.

A doença de Alzheimer é um mal que aflige milhares de pessoas ao redor do mundo. Sendo até o momento incurável e suas causas ainda incompreendidas, a detecção e tratamento precoce se tornaram as maiores aliadas no seu combate. Um método para se detectar a presença da doença é através da análise de imagens de ressonância magnética do cérebro, contudo, ainda depende muito da experiência do examinador. Para isso é treinada uma rede neural convolucional para classificar imagens de ressonância magnética de cérebros com doença de Alzheimer, predizendo não só se o paciente está sadio (*non demented*), mas também em um dos 3 estágios da doença (*mild demented*, *very mild demented* ou *moderate demented*). Seis redes rasas e criadas do zero são treinadas a partir do aumento de dados de um conjunto de dados público, desbalanceado e rotulado de imagens de ressonância magnética de diferentes fatias horizontais do cérebro. Aquela que apresentou melhor acurácia é comparada a uma linha de base consistindo de um classificador (SVM) treinado com um vetor de características extraído de um ResNet-50 usando os mesmos dados de treinamento. A rede rasa final obteve uma acurácia de 78,5% para o conjunto de teste contra 49,80% da linha de base, além de superá-la na classificação de cada uma das 4 classes. Observou-se também que a técnica de aumento de dados mitigou o problema de desbalanceamento das classes e que o *dropout* aumentou em 8,05% na acurácia do conjunto de teste. Assim, foi possível treinar uma rede neural convolucional capaz de predizer o estágio da doença de Alzheimer a partir de imagens de ressonância magnética de cérebros com acurácia maior que a linha de base definida e com uma leve dificuldade de classificação da doença nos seus estágios iniciais (*mild demented* e *very mild demented*), condizendo com a dificuldade enfrentada por médicos em seus diagnósticos.

Palavras-chave: Redes neurais; Doença de Alzheimer, Ressonância magnética.

ABSTRACT

ISHIBE, E. I. **Feature learning for classification of Alzheimer's disease magnetic resonance images.** 2022. 67 f. Trabalho de conclusão de curso (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2022.

Alzheimer's disease is an illness that afflicts thousands of people around the world. Being so far incurable and its causes still not understood, early detection and treatment have become the greatest allies in its fight. One method to detect the presence of the disease is through the analysis of magnetic resonance images of the brain, however, it still depends a lot on the examiner's experience. For this, a convolutional neural network is trained to classify magnetic resonance images of brains with Alzheimer's disease, predicting not only whether the patient is healthy (non demented), but also in one of the 3 stages of the disease (mild demented, very mild demented, or moderate demented). Six shallow networks created from scratch are trained from the data augmentation of a public, unbalanced, labeled dataset of MRI images of different horizontal slices of the brain. The one that showed the best accuracy is compared to a baseline consisting of a classifier (SVM) trained with a feature vector extracted from a ResNet-50 using the same training data. The final shallow network achieved an accuracy of 78.5% for the test set versus 49.80% for the baseline, and outperformed it in classifying each of the 4 classes. It was also observed that the data augmentation technique mitigated the class imbalance problem and that dropout increased the accuracy of the test set by 8.05%. Thus, it was possible to train a convolutional neural network capable of predicting the stage of Alzheimer's disease from brain MRI images with an accuracy greater than the defined baseline and with a slight difficulty in classifying the disease in its early stages (mild demented and very mild demented), consistent with the difficulty faced by physicians in their diagnoses.

Keywords: Neural networks; Alzheimer's disease; Magnetic resonance imaging.

LISTA DE ILUSTRAÇÕES

Figura 1 – Imagens do cérebro obtidas através do exame PET. Na parte superior se tem cérebros de pacientes saudáveis e na inferior com sintomas leves de Alzheimer. Do lado esquerdo, estão os níveis de beta-amiloide e do lado direito da proteína tau	17
Figura 2 - Imagens de ressonância magnética do cérebro de um voluntário usando um scanner de 3 teslas (esquerda) e um de 9,4 teslas (direita).....	19
Figura 3 – Modelo representativo de um neurônio biológico em que se pode ver os dendritos, corpo celular e axônio. Uma seta indica a direção em que o sinal é transmitido	20
Figura 4 - Representação de um neurônio artificial. A saída é o resultado do somatório ponderado das entradas passando por uma função de ativação, que no caso, é a função sigmóide	20
Figura 5 - Representação de uma rede neural artificial em que é mostrado a camada de entrada (em azul escuro do lado esquerdo) as camadas escondidas (em turquesa no centro) e a camada de saída (em azul claro no lado direito)	21
Figura 6 – Ilustração dos problemas de <i>underfitting</i> e <i>overfitting</i> para um problema de classificação binária. Do lado esquerdo, há o problema de <i>underfitting</i> , em que a rede neural é incapaz de aprender o padrão. No centro, ocorreu a generalização. Do lado direito, há o problema de <i>overfitting</i> , em que a rede neural se ajusta excessivamente aos dados de treinamento.....	23
Figura 7 – Exemplo de transformações de imagens. As colunas da esquerda para a direita mostram as seguintes transformações: sem transformação, rotação, borrão, contraste, redimensionamento, iluminação e perspectiva.....	26
Figura 8 – Ilustração do funcionamento de uma convolução. A operação de convolução está sendo realizada entre a matriz a esquerda 7x7 e a matriz central 3x3, resultando na matriz a direita.....	27
Figura 9 - Módulo <i>Inception</i> tradicional	28
Figura 10 – Arquitetura do GoogLeNet. Inicia-se da esquerda para a direita e pode ser observado camadas convolucionais, em azul, camadas de <i>polling</i> , em vermelho, camadas de <i>softmax</i> , em amarelo, e outras camadas (como de concatenação e normalização), em verde.	29
Figura 11 - Arquitetura da ResNet 34. Os blocos azuis (de diferentes tonalidades) indicam as camadas de convolução, os blocos amarelos, camadas de <i>pooling</i> e blocos laranja, camada FC. As linhas sólidas indicam os <i>skipping layers</i> , quando a entrada e saída possuem a mesma dimensão, enquanto que as linhas pontilhadas indicam <i>skipping layers</i> com <i>pooling</i> , quando existe a necessidade de aumentar a dimensão da entrada para igual à da saída.	30
Figura 12 - Estrutura do residual blocks, shortcut connections ou skipping layers. Observa-se que há a soma do vetor de características x ao mesmo valor após a transformação por 2 camadas de pesos.	30
Figura 13 – Arquitetura da MobileNet. Pode-se observar que ela faz uso de <i>depthwise separable convolutions</i> e a <i>pointwise convolutions</i> para então finalizar com uma camada de <i>global average pooling</i> e um camada FC.....	31
Figura 14 – Represetação de uma <i>dense block</i> utilizada na DenseNet, modelo em que o ChexNet é baseado.....	32

Figura 15 – <i>Pipeline</i> da proposta. No lado esquerdo, observa-se que o conjunto de dados divididos em treino e teste, seguido por um bloco de pré-processamento (amarelo). No centro da figura existem dois blocos um de rede rasa (azul), que treina um modelo CNN, e um da linha de base (verde), que treina um classificador utilizando extração de características. Por fim, um bloco de comparação e análise dos resultados.	36
Figura 16 – Distribuição das classes no conjunto de treino (5121 imagens) e no conjunto de teste (1279 imagens).....	39
Figura 17 – Diagrama do processo de extração de características. O conjunto de dados pré-processado é transformado em vetores de características através dos modelos pré-treinados (extratores de características). Cada um dos 4 vetores de características gerado é dado como entrada para os classificadores. A combinação que apresentar melhor acurácia no conjunto de teste será utilizada como <i>baseline</i>	43
Figura 18 – Representação da rede rasa F formada por 7 camadas convolucionais (vermelho), 6 camadas <i>max-pooling</i> (verde) e 1 camada FC (azul).	44
Figura 19 - Representação da rede rasa F com <i>dropout</i> de 15% formada por 7 camadas convolucionais (vermelho), 6 camadas <i>max-pooling</i> (verde) e 1 camada FC (azul). Em amarelo, estão as camadas de <i>dropout</i> do Keras e são aplicadas após cada par convolucionais/ <i>max-pooling</i>	46
Figura 20 - Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) do modelo <i>baseline</i> (ResNet 50 e SVM) avaliado no conjunto de treino	51
Figura 21 – Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) do modelo <i>baseline</i> (ResNet 50 e SVM) avaliado no conjunto de teste	51
Figura 22 – Gráfico da perda por época durante o treinamento das redes rasas sem <i>dropout</i>	52
Figura 23 – Gráfico que mostra a acurácia global pela taxa de <i>dropout</i> para as redes rasas propostas usando o conjunto de teste. Em preto, a melhor acurácia obtida entre as redes rasas sem <i>dropout</i> (rede rasa F) no conjunto de teste.....	54
Figura 24 – Gráfico da perda (esquerda, laranja) e acurácia (direita, azul) em função da época durante o treinamento da rede F com <i>dropout</i> de 15%.....	56
Figura 25 - Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) da rede F com <i>dropout</i> de 15% avaliado no conjunto de teste	57
Figura 26 - Comparação entre acurácia global e de cada classe do modelo <i>baseline</i> e da rede rasa F com <i>dropout</i> de 15%	58

LISTA DE TABELAS

Tabela 1 - Configuração do hardware utilizado na execução do código utilizado no projeto	37
Tabela 2 – Exemplos de imagens de ressonância magnética para cada uma das 4 classes: NonDemented, VeryMildDemented, MildDemented e ModerateDemented presentes em [29]. As podem ser provenientes de fatias horizontais de diferentes posições do cérebro	38
Tabela 3 - Operações de transformação de imagem utilizadas no aumento de dados. Na tabela também é apresentado a variação permitida para cada operação e um exemplo de transformação	40
Tabela 4 - Número de transformações possíveis para 5 tipos de operações, sem considerar repetições ou ordem das operações. Na coluna combinações, separado por vírgulas estão todas as transformações possíveis e entre colchetes está a combinação de operações	41
Tabela 5 – Comparação entre a quantidade esperada de dados antes e depois do aumento de dados	42
Tabela 6 – Redes rasas “do zero” propostas. São formadas por uma sequência de camadas convolucionais seguidas por uma de <i>max-pooling</i> e, ao fim, uma camada convolucional e uma FC.....	45
Tabela 7 – Dimensão do vetor de características após a última camada <i>max-pooling</i> para cada rede rasa.	45
Tabela 8 – Número de operações realizadas por imagem para cada uma das classes. O subtotal A indica a quantidade de imagens geradas pelo aumento de dados por classe e o subtotal B a quantidade de imagens produzidas por número de operações	47
Tabela 9 – Quantidade de operações de transformação realizada pela técnica de aumento de dados por classe.....	48
Tabela 10 – Acurácia obtida do conjunto de teste avaliado no classificador SVM utilizando vetores de características extraídos por diferentes modelos pré-treinados. As 4 primeiras colunas mostram a acurácia para cada uma das classes e a última, uma acurácia global.....	49
Tabela 11 - Acurácia obtida do conjunto de teste avaliado no classificador <i>Random Forest</i> utilizando vetores de características extraídos por diferentes modelos pré-treinados. As 4 primeiras colunas mostram a acurácia para cada uma das classes e a última, uma acurácia global	49
Tabela 12 – Valor mínimo de acurácia entre as classes para cada classificador e modelo pré-treinado. Juntamente com o valor mínimo está a classe associada	50
Tabela 13 - Acurácia obtida do conjunto de treino para cada rede rasa proposta. As 4 primeiras colunas mostram a acurácia para cada um das classes e a última, uma acurácia global. Para o treinamento foram utilizadas as imagens com aumento de dados e regularização e não foi aplicado a técnica de <i>dropout</i>	53
Tabela 14 – Acurácia obtida do conjunto de teste para cada rede rasa proposta. As 4 primeiras colunas mostram a acurácia para cada um das classes e a última, uma acurácia global. Para o treinamento foram utilizadas as imagens com aumento de dados e regularização e não foi aplicado a técnica de <i>dropout</i>	53
Tabela 15 – Acurácia global do conjunto de treino e de teste da rede F para diferentes taxa de <i>dropout</i>	55

Tabela 16- Acurácia da rede F com <i>dropout</i> de 15% avaliado no conjunto de teste e de treino. As 4 primeiras colunas mostram a acurácia para cada um das classes e a última, uma acurácia global.....	55
Tabela 17 - Acurácia da rede rasa F modificada para um problema de classificação binária. A avaliação foi realizada sobre o conjunto de testes para todas as combinações de classes	59

SUMÁRIO

1	INTRODUÇÃO	14
1.1	CONTEXTUALIZAÇÃO	14
1.2	JUSTIFICATIVA E MOTIVAÇÃO	15
1.3	QUESTÕES DE PESQUISA E OBJETIVOS	16
2	REVISÃO BIBLIOGRÁFICA	17
2.1	DOENÇA DE ALZHEIMER	17
2.2	IMAGEM DE RESSONÂNCIA MAGNÉTICA	18
2.3	REDES NEURAIS ARTIFICIAIS	19
2.3.1	<i>Treinamento e teste de uma rede neural artificial</i>	22
2.3.2	<i>Overfitting e underfitting</i>	22
2.3.3	<i>Dropout</i>	23
2.3.4	<i>Métricas</i>	24
2.3.5	<i>Aprendizado profundo</i>	24
2.3.6	<i>Transferência de aprendizado</i>	25
2.3.7	<i>Aumento de dados</i>	25
2.4	REDES NEURAIS CONVOLUCIONAIS	26
2.4.1	<i>GoogLeNet ou Inception</i>	28
2.4.2	<i>ResNet</i>	29
2.4.3	<i>MobileNet</i>	30
2.4.4	<i>CheXNet</i>	31
2.5	ESTADO DA ARTE	32
3	METODOLOGIA E DESENVOLVIMENTO.....	36
3.1	PROPOSTA	36
3.2	CONJUNTO DE DADOS	37
3.3	PRÉ-PROCESSAMENTO	39
3.4	LINHA DE BASE (<i>BASELINE</i>)	42
3.5	REDE RASA “DO ZERO”	44
3.6	COMPARAÇÃO E ANÁLISE DOS RESULTADOS	46
4	RESULTADOS E ANÁLISE	47
4.1	PRÉ-PROCESSAMENTO DE DADOS	47
4.2	LINHA DE BASE (<i>BASELINE</i>)	48
4.3	REDE RASA “DO ZERO”	52
4.4	COMPARAÇÃO E ANÁLISE DOS RESULTADOS	57
5	CONCLUSÕES	60

1 INTRODUÇÃO

1.1 Contextualização

Em 2003, o Prêmio Nobel em Fisiologia ou Medicina foi concedido ao químico Paul C. Lauterbur e ao físico Sir Peter Mansfield devido a descoberta da captura de imagem de ressonância magnética (MRI, do inglês *Magnetic Resonance Imaging*) [1]. Essa técnica, que consiste na produção de campos magnéticos para a detecção de moléculas de água, permitiu a visualização de estruturas e funções do corpo humano em um nível de detalhe nunca observado anteriormente e é considerado um dos avanços mais importantes na medicina no século [2].

Com a teoria desenvolvida por Lauterbur e Mansfield desde 1971 [3], foi possível, em 1989, a patente scanner de MRI por Raymond Damadian [4]. Tal equipamento é capaz de produzir imagens tridimensionais detalhadas dos tecidos e órgãos. E essas imagens, além de proporcionarem um melhor conhecimento do corpo humano, também possuem um papel importante no diagnóstico e tratamento de doenças como na detecção de esclerose múltipla e câncer [1].

É notável a importância do MRI na medicina, assim como apontam os dados de [5], [6] e [7] em que se pode ver que o número de exames e scanners só tem aumentado com o tempo. Uma das aplicações do MRI na medicina é o seu uso em avaliações clínicas de pacientes com suspeita de Alzheimer [8].

O primeiro caso relatado da doença de Alzheimer foi em 1906 pelo psiquiatra clínico e neuroanatomista Louis Alzheimer, que observou sintomas de paranoia, sono progressivo, distúrbio de memória, agressão e confusão em uma mulher de 50 anos. No seu entendimento, baseado no conhecimento da época, ela tinha uma doença severa no seu córtex cerebral [9].

Hoje se descreve a doença de Alzheimer (ou simplesmente, Alzheimer) como uma degeneração progressiva das funções cognitivas do cérebro que avança em três estágios: leve, moderado e grave. Tipicamente a doença se caracteriza por uma perda de memória suave (leve ou *mild*), podendo evoluir para uma necessidade de cuidados especiais (moderado ou *moderate*), até o total declínio das faculdades mentais e físicas (grave ou *severe*) e culminando em sua morte [10] [11] [12]. Causado pela perda dos neurônios e de suas conexões, dentre outras alterações patológicas, o Alzheimer ainda é de difícil detecção e ainda não possui um tratamento efetivo [13] [14] [15].

O Alzheimer afeta mais de 24 milhões de pessoas no mundo desde 2011 e só no Brasil estima-se cerca de 1 milhão de pessoas [16] e, até o momento, o melhor método para retardá-

lo é a sua detecção e tratamento precoce [15] [17] e, devido a isso, o uso de MRI é um procedimento integral nas avaliações clínicas de pacientes com suspeita de Alzheimer [8].

Inspirado pelo funcionamento das células neurais do cérebro humano, em 1943 foi introduzido o primeiro modelo de neurônio artificial por Warren McCulloch e Warren Pitts [18] [19] [20]. Essa ideia e de muitas outras, como o de Hebb [21], Rosenblatt [22] e McClelland e Rumelhart [23], culminou e despertou o interesse de uma nova subárea da inteligência artificial chamada de Redes Neurais Artificiais. As Redes Neurais Artificiais (ANN, do inglês *Artificial Neural Network*), objeto de estudo dessa nova subárea, é um modelo matemático que, de forma muito simplificada, é capaz de “aprender” padrões de um conjunto de dados [19].

Devido a essa capacidade de extrair padrões, as ANN têm exercido uma função fundamental na medicina [24], como na detecção e classificação de patologias. Dentre essas aplicações, entra em destaque a análise e tratamento de imagens médicas, como imagens de raio-X, de tomografias computadorizadas (*CT scan*, do inglês, *Computed tomography scan*), de exame PET (do inglês, *Positron emission tomography*) e de MRI.

As arquiteturas de ANN comumente utilizadas para resolver problemas de análise e tratamento de imagens médicas são as redes neurais convolucionais (CNN, do inglês *Convolutional Neural Network*), que extraem características (*features*) através do encadeamento de filtros convolucionais e processam essas características para a produção do resultado final [24].

Após a breve explicação sobre a importância do MRI na detecção do Alzheimer e o poder das ANN, foi proposto um projeto que aplica técnicas de CNN para detecção da doença a partir de imagens de ressonância magnética do cérebro. Com esse estudo, diferentes arquiteturas de CNN foram avaliadas e os resultados obtidos comparados a desempenho na literatura.

1.2 Justificativa e motivação

O tratamento precoce ainda é a melhor forma de combate ao Alzheimer, uma vez que pode-se retardar os efeitos degenerativos irreversíveis da doença, prolongando a saúde mental do paciente e reduzindo a sobrecarga nos seus cuidadores, geralmente papel ocupado por algum familiar [25] [26].

É natural que, para realizar o tratamento precoce do Alzheimer, seja necessária a sua detecção precoce. O usual diagnóstico clínico consiste em um resultado subjetivo que se baseia em informações do histórico do paciente, testes cognitivos e entrevista com familiares, por isso,

exige uma grande experiência do médico [17] [27]. Somado a isso, em fases iniciais, é difícil a distinção entre o Alzheimer e problemas relacionados com a idade, estresse ou outro distúrbio cerebral [17] [27]. Para ilustrar essa dificuldade da detecção, um estudo realizado em Cambridge mostra que neuropatologistas interpretaram 76% dos cérebros cognitivamente intactos com casos de Alzheimer quando fornecidos com dados clínicos de pacientes idosos [15].

Imagens de ressonância magnética (ou imagens de MR) provêm uma informação adicional para uma melhor acuidade diagnóstica da doença, mas ainda sofre de algumas limitações, por exemplo, dependem enormemente da experiência do examinador [28].

Um método para se analisar essas imagens é através de CNN. Esse algoritmo, após a extração de padrões de diversas imagens de MR do cérebro de pacientes com e sem Alzheimer, possibilitaria a detecção da doença e também do seu estágio. O poder da CNN é tal que ela não só poderia extrair características como o conhecido atrofamento do hipocampo e do córtex, mas também de outras regiões do cérebro em que ainda não temos evidências que contenham informações sobre a doença [28].

Por fim, a CNN é uma ferramenta rápida, eficiente e objetiva que auxiliaria o médico no diagnóstico final do paciente.

1.3 Questões de pesquisa e objetivos

Neste trabalho espera-se criar uma CNN que tem como entrada a imagem de MR do cérebro de uma paciente e gera como saída uma das seguintes classificações: sem traços (*Non Demented*), muito leve (*Very Mild Demented*), leve (*Mild Demented*) ou moderado (*Moderate Demented*). A aplicação e também o treinamento supervisionado farão uso de imagens de MRI de cérebros rotulados disponíveis em [29].

Dado esse objetivo geral, a seguinte questão de pesquisa surge e que ajudará a guiar o desenvolvimento desse projeto:

“Qual algoritmo produz o melhor resultado na classificação do conjunto de imagens de Alzheimer?”

Diante desta questão, alguns objetivos específicos foram definidos:

- Avaliar arquiteturas e técnicas na literatura que contribuam na melhora do desempenho na classificação das imagens de MR;
- Criar um linha de base para comparar o resultado obtido.

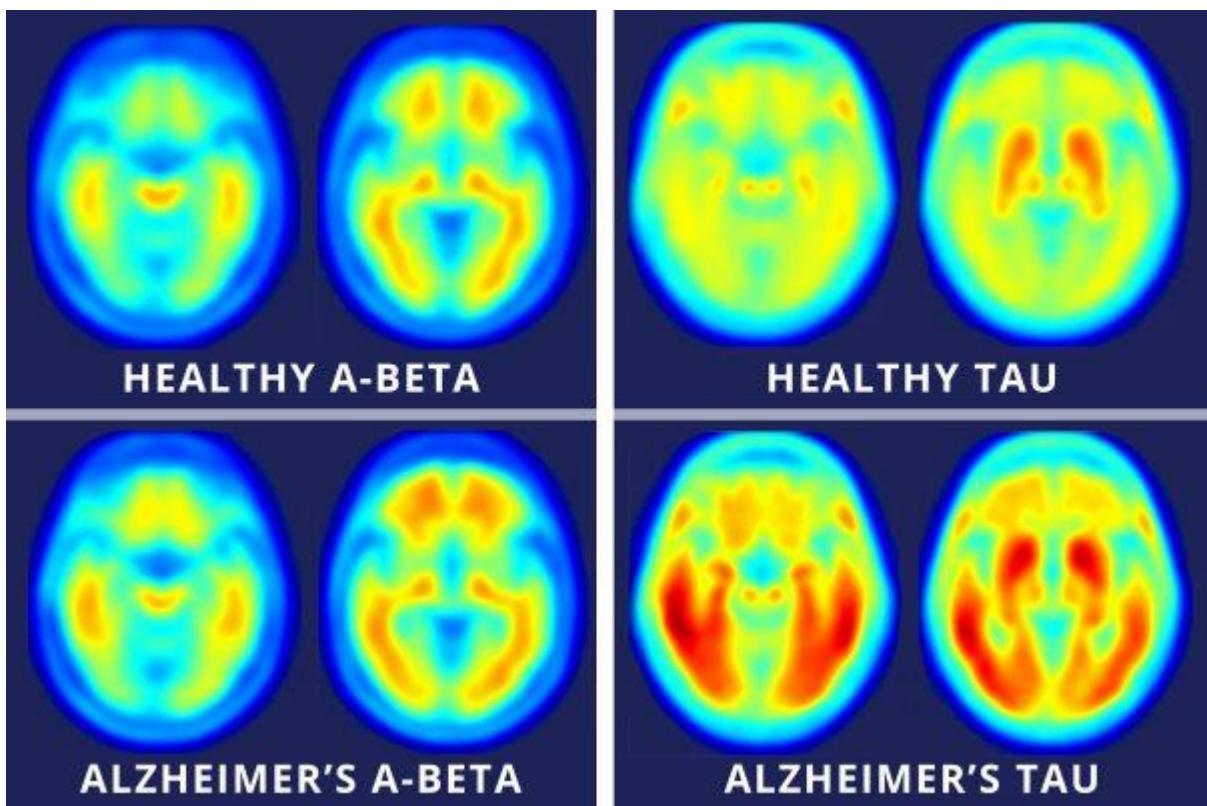
E com isso, espera-se que o algoritmo traga os benefícios mencionados.

2 REVISÃO BIBLIOGRÁFICA

2.1 Doença de Alzheimer

A doença de Alzheimer é um distúrbio neurodegenerativo irreversível acometida primordialmente em idosos e é caracterizada pela progressiva perda de memória e raciocínio. O Alzheimer é também a causa principal da demência, constituindo cerca de 60 a 80% dos casos, e a hipótese mais bem aceitas que o define é o acúmulo do peptídeo beta-amiloide ($A\beta$) e de tranças neurofibrilares de proteína tau (NFT, do inglês, *Neurofibrillary tangles*) [31] [32]. Uma imagem mostrando os níveis dessas proteínas no cérebro de um paciente saudável e um com Alzheimer é mostrado na Figura 1.

Figura 1 – Imagens do cérebro obtidas através do exame PET. Na parte superior se tem cérebros de pacientes saudáveis e na inferior com sintomas leves de Alzheimer. Do lado esquerdo, estão os níveis de beta-amiloide e do lado direito da proteína tau



Fonte: [33].

A neurotoxina $A\beta$ é gerada pela quebra proteica da sequência chamada de proteína precursora de amiloide, conhecida por APP (do inglês, *Amyloid Precursor Protein*). O APP é uma membrana glicoproteica que tem um papel importante em uma variedade de atividades

biológicas, como desenvolvimento neuronal, envio de sinais e transporte intracelular e é constituinte de um grande número de neurônios. A razão do depósito do peptídeo A β no cérebro de idosos, especialmente na região do hipocampo e córtex entorrinal, é ainda desconhecido [34] [35] [36].

A proteína tau tem por função de dar forma e estabilidade aos microtúbulos neurais, mas que em pacientes que sofrem de Alzheimer ela se acumula de forma anormal, formando um emaranhado que chamamos de NFT. Esse emaranhado não só compromete a estabilidade dos microtúbulos, mas também leva ao colapso de todo o neurônio que tem seus neurotransmissores comprometidos e param de funcionar [37] [38].

Estudos em organismos mais simples mostram que o Alzheimer, quando detectado em fases iniciais, pode ser parado, postergado ou mesmo revertido, levando a um esperança no combate à doença. Para isso, é preciso de biomarcadores eficientes que possam indicar a presença do distúrbio. A medição dos níveis das proteínas A β e tau no fluido cefalorraquidiano (CSF, do inglês *Cerebrospinal fluid*) e a detecção através de imagens do cérebro são dois biomarcadores importantes e estudados da doença de Alzheimer [39] [40] [41].

2.2 Imagem de ressonância magnética

O MRI é um exame de imagem não invasivo capaz de mostrar com definição as estruturas internas dos órgãos, sendo importante para detectar, diagnosticar e monitorar o tratamento de doenças com a maior brevidade possível, a fim de mitigar ou eliminar os agravos destas. Dada a alta capacidade de diferenciar tecidos, o espectro de aplicações se estende a todas as partes do corpo humano e explora aspectos anatômicos e funcionais [42] [43].

Para o exame, utiliza-se uma grande máquina (scanners de MRI), que produz pulso de campo magnético capaz de provocar o alinhamento (ou magnetização) dos prótons de hidrogênio. Com o fim do pulso, há o retorno do vetor de magnetização desses prótons, em um processo chamado relaxação, e consequente geração de um sinal de radiofrequência. E através de uma sequência de pulsos e leitura por uma antena receptora é feita a aquisição da imagem do corpo humano [42] [43].

As imagens produzidas através da MRI se apresentam em uma escala que varia do branco ao preto, como os cérebros da Figura 2. Os sinais de muita intensidade apresentam-se na cor branca, já os sinais de baixa intensidade apresentam-se na cor preta, sinais intermediários são vistos em escala de cinza. Essas tonalidades representam os mais variados tecidos do corpo, por exemplo, o tecido adiposo fornece o sinal mais intenso devido à interação dos átomos de

hidrogênio presentes nos ácidos graxos, enquanto que o tecido ósseo não apresenta sinal [42] [43].

Figura 2 - Imagens de ressonância magnética do cérebro de um voluntário usando um scanner de 3 teslas (esquerda) e um de 9,4 teslas (direita)



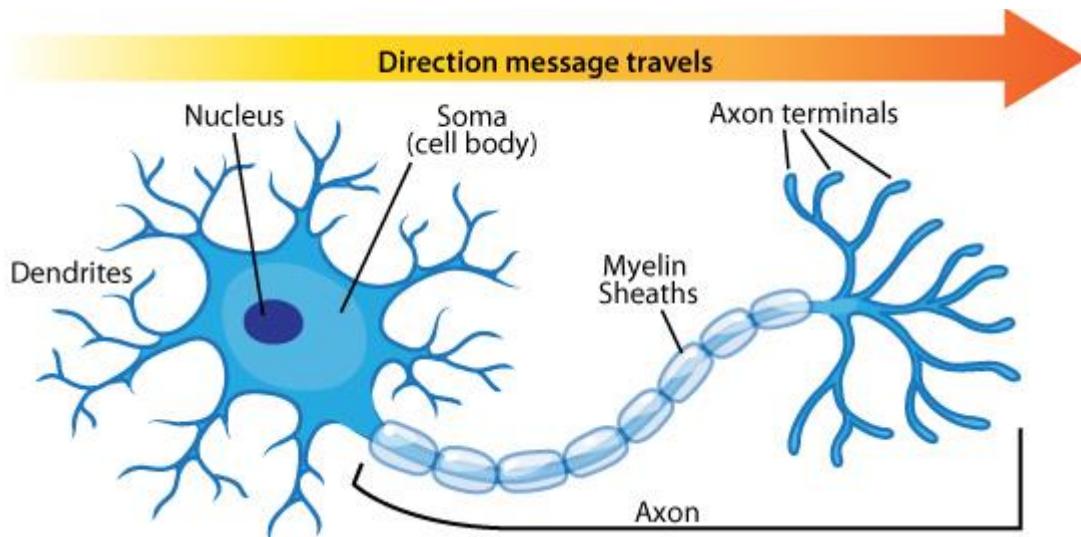
Fonte: [44]

2.3 Redes neurais artificiais

O neurônio biológico é uma célula que recebe sinais bioquímicos através de uma rede de fibras nervosas chamada de dendritos. Esses sinais podem aumentar ou reduzir o potencial elétrico da do corpo celular que, se atingido certo limiar, dispara um sinal pelo seu axônio [45].

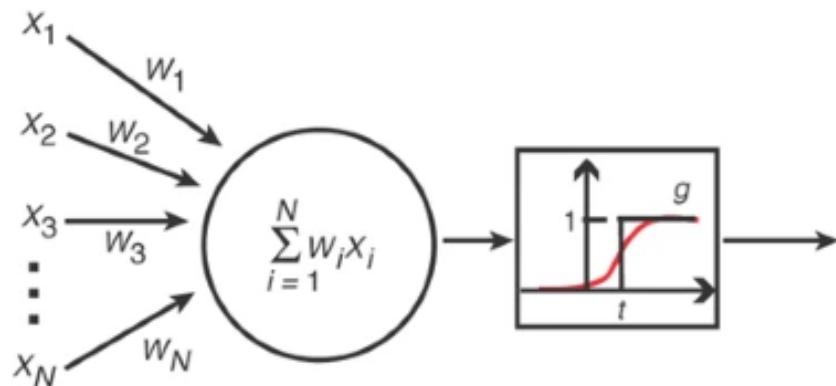
Analogamente ao dendrito, corpo celular e axônio da célula nervosa biológica, como ilustrado na Figura 3, o neurônio artificial, elementos básicos de uma rede neural artificial, possui entrada, função de ativação e saída. Esses neurônios artificiais (ou simplesmente neurônios ou nós) fazem a soma ponderada de sua entrada e o resultado é avaliado em uma função de ativação antes de ser enviado para a saída [46]. A Figura 4 mostra uma representação do neurônio artificial, em que x_1 a x_N são as entradas, w_1 a w_N são os pesos, o somatório ponderado é mostrado dentro do círculo e a função de ativação no retângulo.

Figura 3 – Modelo representativo de um neurônio biológico em que se pode ver os dendritos, corpo celular e axônio. Uma seta indica a direção em que o sinal é transmitido



Fonte: [47]

Figura 4 - Representação de um neurônio artificial. A saída é o resultado do somatório ponderado das entradas passando por uma função de ativação, que no caso, é a função sigmóide



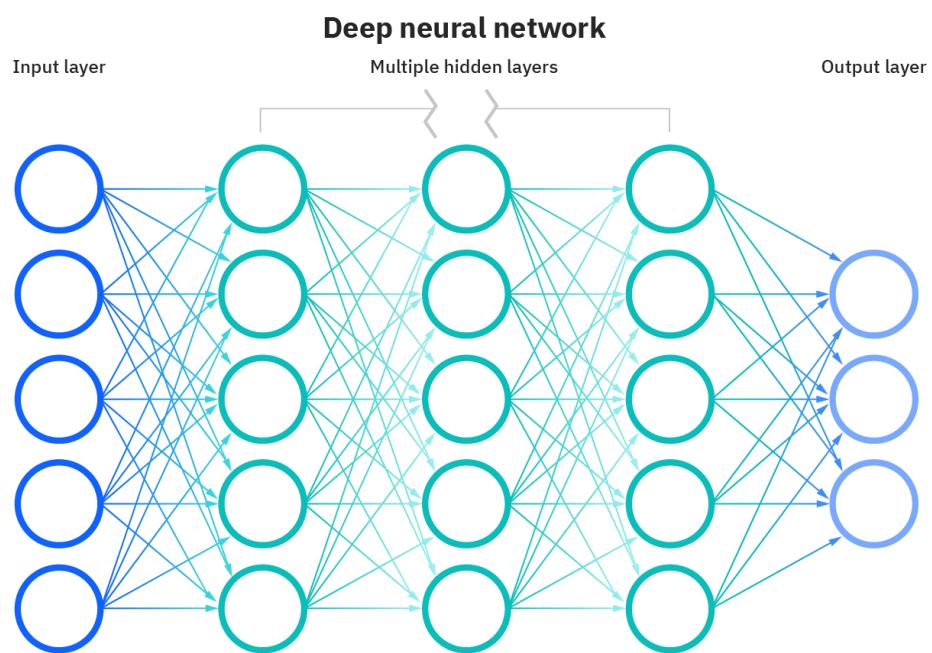
Fonte: [46]

A respeito da função de ativação, ela confere a não linearidade ao modelo, permitindo assim com que os complexos padrões existentes no mundo real possam ser aprendidos [48]. As funções de ativação mais utilizadas são: sigmóide, tangente hiperbólica e a ReLU (do inglês, *Rectified Linear Unit*) [74].

Assim como denominado por Rosenblatt [21], esse modelo inspirado no neurônio biológico é o perceptron, embora o termo “neurônio” também seja usado de forma

intercambiável. A organização desses perceptrons (ou neurônios) em camadas, contendo uma de entrada, uma ou mais escondida e uma de saída, como ilustrado na Figura 5, é o que denominamos redes neurais artificiais ou MLP (do inglês, *Multilayer Perceptron*). Essas camadas são conectadas umas às outras de forma que a saída de todos os neurônios de uma camada é conectada a todas as entradas dos neurônios da camada adjacente seguinte.

Figura 5 - Representação de uma rede neural artificial em que é mostrado a camada de entrada (em azul escuro do lado esquerdo) as camadas escondidas (em turquesa no centro) e a camada de saída (em azul claro no lado direito)



Fonte: [49]

Na MPL, os dados externos são inseridos na camada de entrada e o resultado é produzido na camada de saída. Entretanto, para que esse resultado seja satisfatório para aplicação o qual a MLP foi projetada é necessário que a arquitetura e os pesos das somas ponderadas estejam ajustados.

Em uma MLP, o número de camadas escondidas e número de neurônios em cada uma delas, bem como a função de ativação utilizada são definidos pelo projetista, que deve utilizar do seu conhecimento do problema, da sua experiência e de testes realizados para a decisão [50].

Em tarefas de classificação a MLP possui uma camada final chamada de *softmax*. O *softmax* indica a probabilidade do resultado da MLP de ser de uma determinada classe e, para

isso, conta com 1 neurônio para cada uma das possíveis classificações da tarefa. Cada um desses neurônios possui valores não negativos e a soma de todos eles é 1, logo, a classe resultante é aquela cujo neurônio apresenta o maior resultado (ou maior probabilidade) [51].

2.3.1 Treinamento e teste de uma rede neural artificial

Os pesos das somas ponderadas dos neurônios são ajustados em um processo chamado de treinamento (ou aprendizado), o qual se usa um algoritmo chamado *backpropagation*. Durante o treinamento é apresentado a MLP um conjunto de dados de entrada e suas respectivas saídas esperadas com o intuito de “ensinar” o padrão desse conjunto [46]. Mais detalhadamente, para cada um dos dados do conjunto:

- Calcula-se o resultado produzido pela MLP usando os seus pesos correntes (em primeiro momento, esse pesos podem ser inicializados aleatoriamente);
- Com o resultado produzido pela MLP e o resultado esperado é calculado um erro usando uma função de erro ou função de custo;
- O erro é propagado da última camada para a primeira ajustando os pesos da MLP através de um algoritmo chamado de *gradient descent* [52].

O *gradient descent* visa minimizar a função de erro indicando quais e quanto cada um dos diversos pesos da MLP devem ser ajustados [52].

Para se avaliar o resultado do treinamento, em outras palavras, o poder de generalização da MLP, usa-se um conjunto de dados diferente do utilizado durante o aprendizado. Esse novo conjunto de dados é processado pela MLP já ajustada e o resultado obtido, juntamente com o resultado do treinamento, é utilizado para tirar conclusões acerca da arquitetura.

É papel do projetista, portanto, não só definir como esse treinamento e teste serão realizados, mas também realizar alterações no número de camadas ou neurônios para evitar problemas com o *overfitting* e o *underfitting* [50].

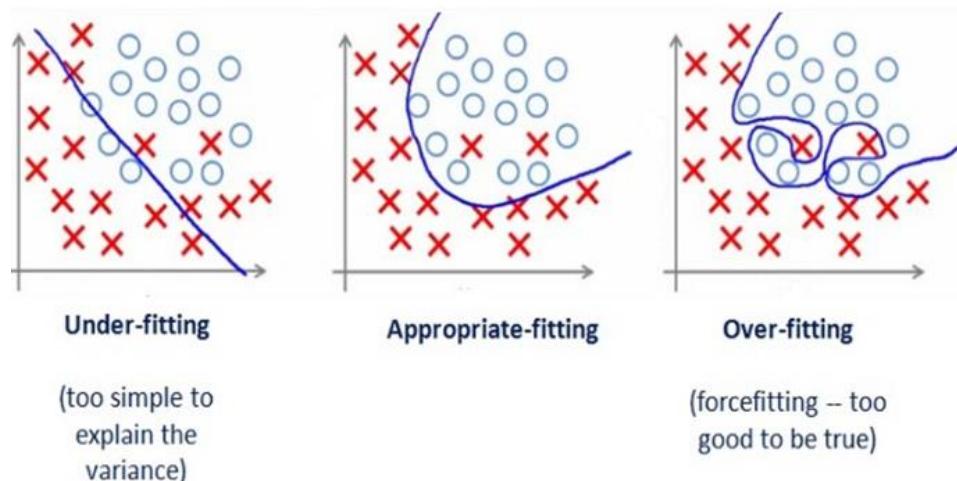
2.3.2 *Overfitting* e *underfitting*

Overfitting é um problema em que a arquitetura escolhida ajusta seus pesos especificamente para o conjunto de dados de aprendizado, sendo incapaz de generalizar o problema e, portanto, incapaz de avaliar novas amostras de dados. Geralmente acontece quando o número de parâmetros é muito maior que o conjunto de dados. Existem algumas técnicas para reduzir o *overfitting* como o uso de regularização, *dropout* e uso de métodos bayesianos [46] [53].

Underfitting, por outro lado, é quando a arquitetura não possui complexidade suficiente para aprender o padrão [53]. Assim como o *overfitting*, ela é incapaz de avaliar novas amostras de dados, mas também é incapaz de avaliar o próprio conjunto de treinamento [54]. Uma maneira de contornar o *underfitting* é aumentando o número de camadas ou de neurônios, ou seja, aumenta a complexidade da MLP.

Uma ilustração sobre os problemas de *overfitting* e *underfitting* é apresentada na Figura 6.

Figura 6 – Ilustração dos problemas de *underfitting* e *overfitting* para um problema de classificação binária. Do lado esquerdo, há o problema de *underfitting*, em que a rede neural é incapaz de aprender o padrão. No centro, ocorreu a generalização. Do lado direito, há o problema de *overfitting*, em que a rede neural se ajusta excessivamente aos dados de treinamento.



Fonte: [55]

2.3.3 Dropout

Assim como o título do artigo em que a técnica foi proposta [56], o *dropout* é uma maneira simples de prevenir redes neurais do *overfitting*. Para isso, ela conta com uma taxa que aleatoriamente desabilita diferentes neurônios e suas conexões durante o treinamento. A intuição por trás dessa técnica é impedir a co-adaptação complexa dos neurônios, fazendo com que eles não dependam do resultado de outros neurônios e, portanto, forçados a contribuir com o resultado final [56] [57]. Desta forma tendem a reduzir o valor dos pesos, assim como a regularização, para tornar a rede mais robusta.

A técnica pode ser empregada com diferentes taxas para diferentes camadas e tipicamente não é realizada na camada de entrada e somente durante a fase de treinamento.

2.3.4 Métricas

Os resultados de treinamento e teste são baseados em métricas. As métricas não só guiam e ajudam no desenvolvimento de uma MPL, mas também são ferramentas para comparação entre diferentes arquiteturas.

Uma métrica importante é na classificação binária é acurácia, e ela representa porcentagem de classificações corretas [58]. É dada por:

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

Onde:

TN: Verdadeiro negativo;

TP: Verdadeiro positivo;

FP: Falso positivo;

FN: Falso negativo.

Analogamente, pode-se obter a acurácia para o caso da classificação multiclasse, assim como apresentado em [58] [59], ela contabiliza a porcentagem de acerto para todas as classes.

2.3.5 Aprendizado profundo

Aprendizado profundo ou DL (do inglês, *Deep Learning*) são arquiteturas que possuem muitas camadas [60] [61] e a MLP é um exemplo típico [62], por isso na literatura também são chamados de redes neurais profundas (DNN, do inglês, *Deep Neural Networks*).

Esse tipo de arquitetura profunda é capaz de resolver problemas complexos e alcançar grandes avanços na resolução de problemas que a comunidade de inteligência artificial tem tentado a anos [63], contudo, o treinamento de redes profundas exige muitos dados e poder computacional [64] [65].

Outras dificuldades impostas pelas DL são os *vanishing* e *exploding gradients*. O *vanishing gradient* é um problema em que, conforme os pesos vão sendo ajustados das camadas mais para as menos profundas, o valor de ajuste (ou o gradiente) fica menor e menor fazendo com que os pesos das camadas iniciais fiquem praticamente inalterados e, por conseguinte, impedindo a convergência da DL [66]. O problema de *exploding gradient*, por outro lado, faz com que o gradiente seja acumulado ficando cada vez maior, levando a DL a divergir do resultado [66].

2.3.6 Transferência de aprendizado

A grande quantidade de camadas de uma DNN, se por um lado traz grandes conquistas na realização de tarefas complexas, por outro, traz dificuldades no treinamento. Um dos problemas é na aquisição de dados, pois para que a arquitetura seja capaz de extrair todas as características do problema a ser resolvido, ela precisa ser treinada com um grande volume de dados e, muitas vezes, eles são raros, caros de serem coletados ou rotulados ou simplesmente inacessíveis [67]. Outro problema é o custo computacional para treinar essa arquitetura, uma vez que algoritmos de treinamento, como o *backpropagation*, são lentos e milhares de parâmetros precisam ser ajustados.

Neste contexto, surge a transferência de aprendizado (*transfer learning*). Essa técnica permite que se utilize arquiteturas (ou modelos) já treinados com grandes volumes de dados (como a ImageNet[68]) para resolver tarefas de um domínio diferente [30].

Para adaptar esses modelos pré-treinados para executar uma tarefa diferente, podemos nos concentrar em 2 métodos: por ajuste fino (*fine-tuning*) e por extração de características (*feature extraction*).

O método por ajuste fino mais comum consiste em treinar o modelo pré-treinado no conjunto de dados da tarefa que se deseja aprender, permitindo somente a alteração dos parâmetros das camadas mais profundas.

No método de extração de características, o resultado (saída) de uma camada arbitária do modelo pré-treinado é utilizado como entrada em uma nova arquitetura.

Os modelos pré-treinados comumente utilizados em artigos científicos na literatura são a VGG, GoogLeNet e a ResNet [69].

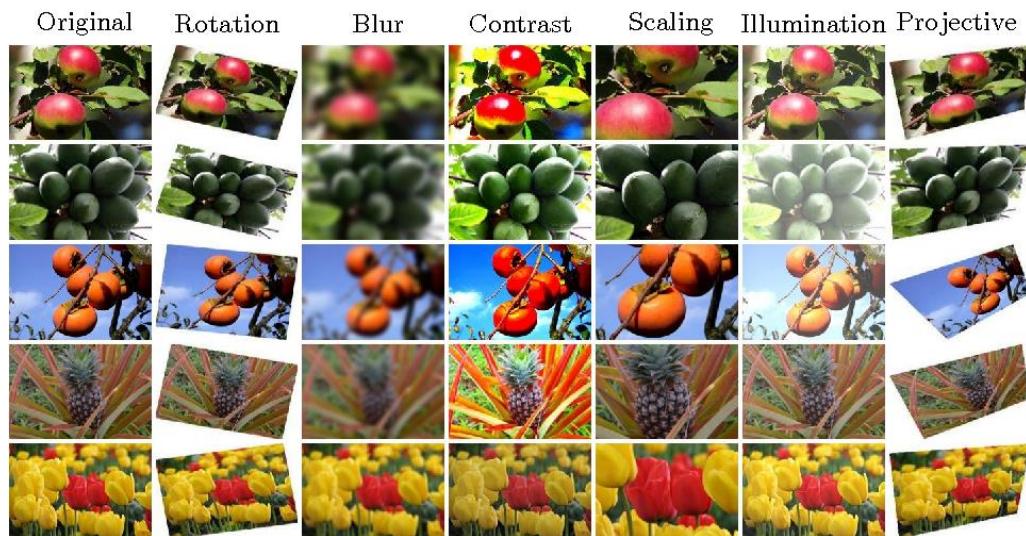
2.3.7 Aumento de dados

O aumento de dados (*data augmentation*) trata-se de um série de métodos que ampliam artificialmente o tamanho do conjunto de treinamento de uma DL. Além do aumento do volume de dados, a técnica também traz como benefício a melhoria na capacidade de generalização dos modelos e o combate ao desbalanceamento de classes [70]. Esses benefícios são desejados em problemas que possuem escassez ou diversidade insuficiente de dados, como é o caso de imagens médicas [71].

Esses métodos de aumento de dados consistem em utilizar os dados originais para artificialmente criar novos exemplos. Para o caso de imagens, pode se utilizar transformação de imagens como rotações, redimensionamento ou alteração na coloração que são técnicas

muito conhecidas e utilizadas na área de visão computacional [71], assim como exemplificado na Figura 7.

Figura 7 – Exemplo de transformações de imagens. As colunas da esquerda para a direita mostram as seguintes transformações: sem transformação, rotação, borrão, contraste, redimensionamento, iluminação e perspectiva



Fonte: [72]

Para acompanhar o progressivo aumento do número de parâmetros de treinamento dos modelos, cria-se a necessidade de conjunto de dados cada vez maiores e, por isso, aumento de dados passou a se tornar uma técnica fundamental e é utilizada em quase todos os modelos no estado da arte [73].

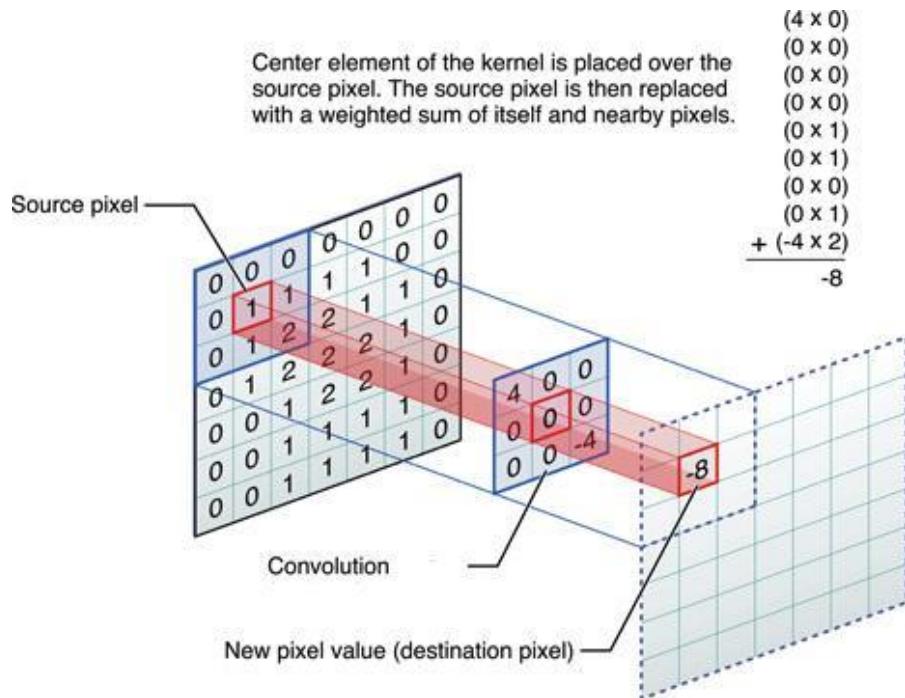
2.4 Redes neurais convolucionais

A CNN (ou ConvNets) é um tipo de ANN que tem ganhado grande destaque na última década, sendo uma das ferramentas mais poderosas especialmente em aplicações de visão computacional [74]. Esse modelo foi nomeado devido a uma operação linear matemática entre matrizes denotada convolução, embora, estritamente falando, a operação realizada é a correlação cruzada [75]. Mas, o termo convolução será mantido devido sua popularidade dentro da subárea.

Limitando-se ao caso de uma convolução de duas matrizes bidimensionais finitas, uma vez que esse trabalho lida com classificação de imagens, podemos entender a operação como o processo de mover uma matriz sobre a outra (em ambas dimensões), calculando a soma dos

produtos ponto a ponto a fim de gerar uma terceira matriz. Podemos ver uma ilustração do processo na Figura 8.

Figura 8 – Ilustração do funcionamento de uma convolução. A operação de convolução está sendo realizada entre a matriz a esquerda 7x7 e a matriz central 3x3, resultando na matriz a direita.



Fonte: [76]

Os blocos básicos de uma CNN são: camadas convolucionais, camadas de funções de ativação, camadas de *pooling* e camadas totalmente conectadas (FC, do inglês, *fully-connected*) [30].

A camada convolucional é composta por um conjunto de matrizes de pesos, chamados de filtros ou *kernels*, em que será aplicada a operação de convolução sobre a matriz de entrada. A intuição por trás dessa camada é permitir que o modelo busque por padrões ou combinações em uma vizinhança ao invés da matriz de entrada inteira, isso torna as CNN mais eficientes que uma ANN tradicional em problemas que envolvem imagens [74].

As camadas de funções de ativação, assim como em uma ANN, conferem a não linearidade ao modelo e são geralmente utilizadas após as camadas convolucionais ou as camadas FC [30]. Em CNNs a função utilizada com maior frequência é a ReLU.

A camada de *pooling* são geralmente aplicadas após algumas camadas convolucionais e tem por finalidade fazer a redução da dimensionalidade do conjunto de matrizes. Entretanto,

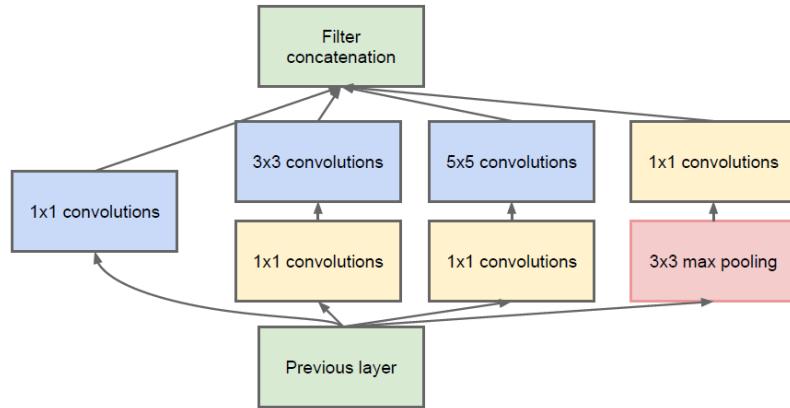
há estudos a favor do abandono dessa camada [30]. Existem diferentes tipos como: o *average-pooling*, o *global average-pooling* e, o mais utilizado, o *max-pooling*.

A camada FC é similar a uma camada escondida de uma ANN fazendo com que todas as saídas da camada anterior a ela sejam conectadas a suas entradas [30]. A camada FC formam as últimas camadas de uma CNN e geralmente compreendem a maior parte dos parâmetros [77] [78].

2.4.1 GoogLeNet ou Inception

Vencendo o VGGNet no desafio “ImageNet Challenge 2014” [82] esse modelo chamou atenção por utilizar módulos chamados de Inception [30]. Esses módulos fazem a concatenação do resultado de diferentes sequências de convoluções de filtros pequenos, assim como ilustrado na Figura 9 [30].

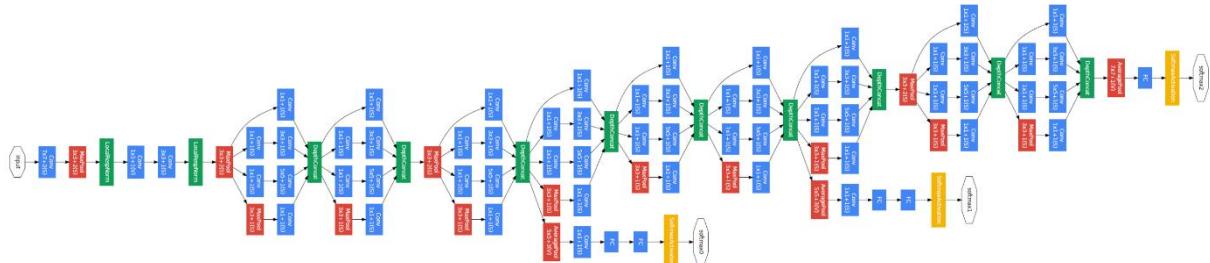
Figura 9 - Módulo *Inception* tradicional



Fonte: [83]

De acordo com [83], essa arquitetura melhora a utilização dos recursos computacionais dentro da rede, permitindo um aumento da sua profundidade mantendo o custo de processamento constante. Mostrada na Figura 10, a GoogLeNet possui 22 camadas com parâmetros (27 se contar as camadas de *pooling*) e apresentou um erro top-5 de 6,67% [83].

Figura 10 – Arquitetura do GoogLeNet. Inicia-se da esquerda para a direita e pode ser observado camadas convolucionais, em azul, camadas de *polling*, em vermelho, camadas de *softmax*, em amarelo, e outras camadas (como de concatenação e normalização), em verde.



Fonte: [83]

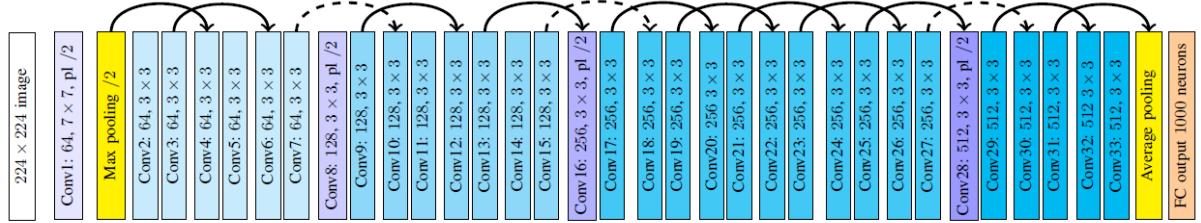
A GoogLeNet mais tarde recebe melhorias produzindo as arquiteturas Inception, tal como a Inception V3, apresentado em [82] [30].

2.4.2 ResNet

A ResNet ou *Residual Networks* é uma rede profunda vencedora da tarefa de classificação da ILSVRC 2015 [84]. Possuindo de 34 a 152 camadas, e de complexidade inferior ao da VGGNet [84], a ResNet alcançou um erro de top-5 de 3,57% que supera o desempenho de um humano sobre o conjunto de imagens em que foi testado [85].

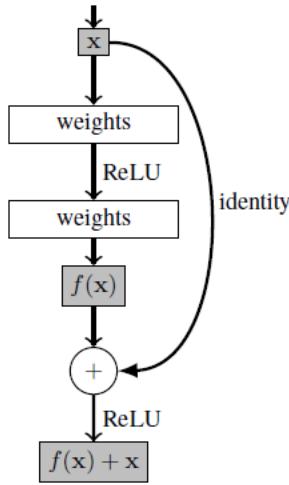
Sua arquitetura, ilustrada na Figura 11, foi possível devido a estruturas chamadas de blocos residuais (*residual blocks*) que conectam 2 camadas de profundidades diferentes. Através dessa conexão são preservadas características da camada de entrada (menos profunda) na camada de saída (mais profunda), por isso também é chamada de *shortcut connection* ou *skipping layer*. Como ilustrado na Figura 12, a *skipping layer* soma o vetor de características x ao seu valor após a transformação por uma ou mais camadas com pesos $f(x)$, resultando em $f(x)+x$.

Figura 11 - Arquitetura da ResNet 34. Os blocos azuis (de diferentes tonalidades) indicam as camadas de convolução, os blocos amarelos, camadas de *pooling* e blocos laranja, camada FC. As linhas sólidas indicam os *skipping layers*, quando a entrada e saída possuem a mesma dimensão, enquanto que as linhas pontilhadas indicam *skipping layers* com *pooling*, quando existe a necessidade de aumentar a dimensão da entrada para igual à da saída.



Fonte: [30]

Figura 12 - Estrutura do residual blocks, shortcut connections ou skipping layers. Observa-se que há a soma do vetor de características x ao mesmo valor após a transformação por 2 camadas de pesos.



Fonte: [30]

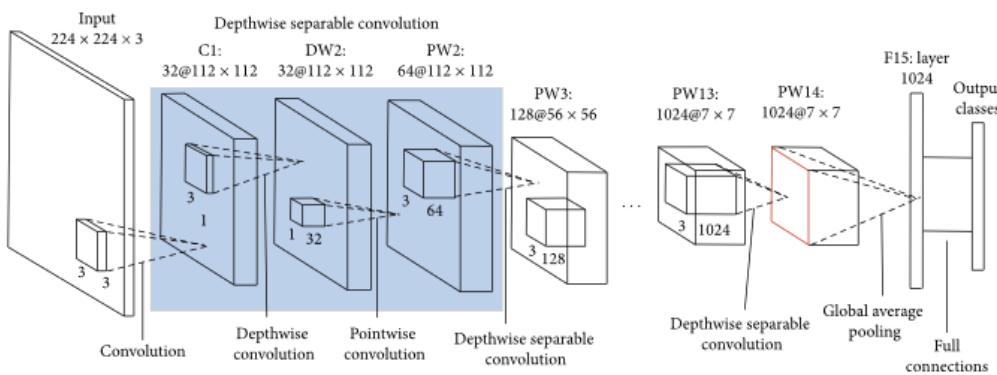
Essa abordagem da ResNet permite que se construa arquitetura mais profundas, que sofrem do problema de *vanishing* ou *exploding gradients* [84].

2.4.3 MobileNet

O MobileNet é uma CNN de código aberto disponibilizada pelo Google desenvolvida para ser utilizada em aplicações móveis e embarcadas, que tem como característica possuírem recursos limitados, como processamento, memória e bateria. Ela faz uso de convoluções separáveis por profundidade (em inglês, *depthwise separable convolutions*) que reduzem significantemente o número de parâmetros treináveis se comparado a uma CNN tradicional de mesma profundidade. Essa convoluções separáveis por profundidade faz a operação de

convolução para cada canal separadamente e depois combina seus resultados por meio de um filtro 1x1 que foi chamada de convolução por ponto (em inglês, *pointwise convolution*), assim, fazendo com que o modelo seja computacionalmente muito eficiente a um custo de uma pequena perda na acurácia se comparado a uma Inception V3, por exemplo [86] [87] [88]. A MobileNet pode ser vista na Figura 13.

Figura 13 – Arquitetura da MobileNet. Pode-se observar que ela faz uso de *depthwise separable convolutions* e a *pointwise convolutions* para então finalizar com uma camada de *global average pooling* e um camada FC.



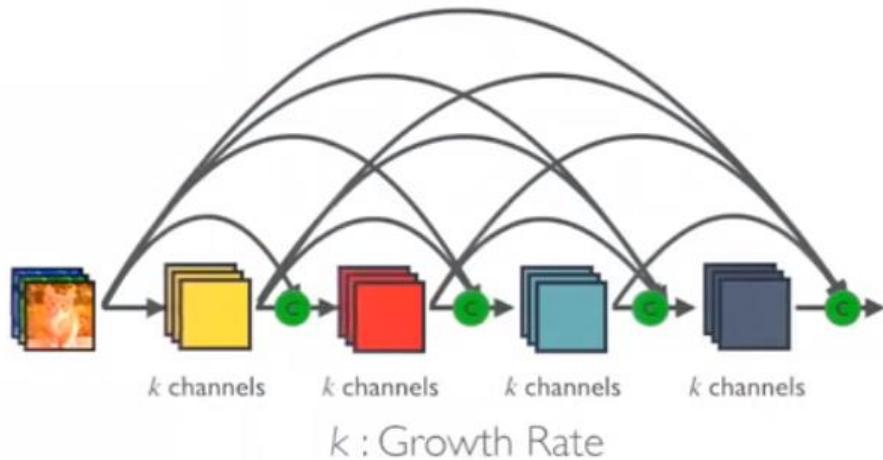
Fonte: [87]

2.4.4 CheXNet

A CheXnet é uma CNN desenvolvida para detecção de pneumonia a partir de imagens de raio-X do tórax e apresentou níveis de acerto que excedem radiologistas. O modelo tem como entrada imagens de raio-X do tórax e retorna a probabilidade do paciente ter pneumonia juntamente, com um mapa de calor localizando a região da imagem que mais apresenta o indicativo da doença. Fora treinada usando o conjunto de dados ChestX-ray14 que possui 112.120 imagens de raio-X frontais do peito, individualmente rotulado com até 14 diferentes tipos de doenças torácicas [89].

O modelo consiste em uma DenseNet [90] de 121 camadas e tem como entrada imagens de 224 x 224 pixels. A DenseNet é formada por *dense blocks* que é formada por uma camada de convolução, função de ativação ReLU e *Batch Normalization* e, diferente das arquiteturas convencionais, nelas cada uma das camadas recebe adicionalmente as entradas de todas as camadas que a precedem e propaga seu mapa de características para todas as camadas que a sucedem, encorajando o reuso de padrões. Uma imagem ilustrativa da *dense block* é apresentada na Figura 14.

Figura 14 – Represetação de uma *dense block* utilizada na DenseNet, modelo em que o ChexNet é baseado.



Fonte: [90]

2.5 Estado da arte

Diversos são os trabalhos publicados que buscam resolver o problema de detecção da doença de Alzheimer a partir de imagens de MR, em especial, de pacientes nos estágios iniciais chamados de MCI (do inglês, *Mild Cognitive Impairment*) que denominamos de muito leve.

Em [91], uma arquitetura VGG 16 pré-treinada foi utilizada para classificação binária entre pacientes com ou sem Alzheimer. Nesta arquitetura foi inserida uma camada final FC e todos os seus pesos foram retreinados usando o conjunto de imagens, que foi pré-processado através de normalização de intensidade (*intensity normalization*) e segmentação do crânio (*skull stripping*). Neste retreinamento, que inicia com os pesos já treinados na ImageNet, foi utilizado o otimizador *stochastic gradient descent* com o Nesterov Momentum, que, de acordo com os autores, é uma combinação que oferece um bom desempenho para imagens médicas, por facilitar a busca pelo mínimo global. O resultado de [91] foi uma acurácia de 99% sobre o conjunto de imagens de teste e um mapa de calor para visualização das áreas afetadas pelo cérebro usando o Grad-CAM [92].

O trabalho de [93] faz a seleção de imagens transversais de maior entropia do conjunto de dados OASIS-3 (do inglês, *Open Access Series of Imaging Studies*) [94]. Segundo os autores, essas imagens de maior entropia podem ser consideradas as mais informativas e que provêm maior robustez no treinamento, portanto, são suficientes no treinamento. Essa técnica foi utilizada no treinamento de duas arquiteturas pré-treinadas, a VGG 16 e a Inception V4, para a classificação entre pacientes com ou sem Alzheimer. Somente reajustando a última camada de

ambas as arquiteturas obteve-se uma acurácia média de 92,3% na VGG 16 e 96,25% na Inception V4 com um conjunto de treinamento de somente 5120 imagens.

As arquiteturas LeNet e GoogLeNet foram usadas para verificação da presença ou não da doença de Alzheimer em [95]. Para ambos modelos de classificação, foi utilizado o conjunto de imagens da ADNI (do inglês, *Alzheimer's Disease Neuroimaging Initiative*) e um pré-processamento extensivo e único. Embora o desbalanceamento seja algo que pode mudar o resultado final, [95] mostra que os modelos são robustos suficientes, sendo quase invariante a esse problema. A melhor acurácia atingida foi de 98,79% para o LeNet e de 98,84% para o GoogLeNet.

A AlzNet (do inglês, *Alzheimer Network*) é uma CNN proposta em [97] que faz a classificação binária de pacientes de Alzheimer através de imagens de MR. Essa arquitetura é composta por 5 camadas convolucionais, cada uma seguida por uma camada de *max-pooling*, uma camada FC com *dropout* e uma camada final com somente 1 neurônio usando função de ativação sigmoidal para realizar a classificação. Ela foi treinada com um total de 15200 imagens extraídas de 240 pacientes (170 com Alzheimer e 70 sem) presentes no banco de dados OASIS-3 [94] e resultou em uma acurácia de 97,88% no conjunto de treinamento e 99,30% no de teste.

Em [98] são apresentadas diversas propostas de arquiteturas para detecção da doença de Alzheimer utilizando o banco de imagens OASIS [94]. Essas propostas consistem basicamente na: BrainNet2D, BrainNet3D e ResNet18. A primeira, formada por 4 camadas convolucionais 2D, cada uma seguida por uma camada de *max-pooling* 2D, e uma camada de saída, tem como entrada 10 fatias centrais da imagem 3D do cérebro e apresenta uma saída para todo o conjunto. A segunda é formada por 5 pares de camadas convolucionais 3D e de *max-pooling* 3D, uma camada de *global average-pooling* e uma camada de saída e, diferentemente da BrainNet2D, recebe como entrada a imagem 3D (176 fatias totais). A última consiste em um arquitetura pré-treinada em que aplicado o *fine-tuning* e tem como entrada uma fatia (imagem) do cérebro que produz uma única saída. Para cada uma das propostas de [90] são apresentadas variações que permitem a classificação de duas (saudável e doente) e três classes (saudável, estágio leve, estágio avançado) e fazem uso de *Batch Normalization*¹ e *Cyclical learning rate*². Além disso, o trabalho faz uso da métrica acurácia balanceada (*Balanced Accuracy*) para lidar com o

¹ *Batch normalization* é uma técnica de treinamento da rede neural profunda que padroniza as entradas para uma camada a cada *mini-batch* (mini-lote), melhorando o desempenho do modelo [98]. *Mini-Batch Gradient Descent* é uma variação do algoritmo de *Gradient Descent* que divide o conjunto de entrada em pequenos lotes (*mini-batches*) que são utilizados na atualização dos pesos, isso traz um ganho computacional ao processo [99].

² *Cyclical learning rate* é uma estratégia que permite a oscilação da taxa de aprendizado entre dois valores [98].

problema de desbalanceamento de dados. O melhor resultado obtido foi da ResNet18 com um porcentagem de 93% tanto para acurácia como para a acurácia balanceada para a classificação binária e uma acurácia de 89% e acurácia balanceada de 88% para a classificação multiclasse.

Uma análise da robustez de uma CNN para classificação de pacientes com e sem Alzheimer de imagens de MR de duas populações de etnia e nível de educação diferente é apresentada em [100]. Uma população é obtida da ADNI [96] em que 83,59% são caucasianos e a maioria possui alto nível de educação, enquanto que a outra é obtida da SNUBH (do inglês, *Seoul National University Bundang Hospital*) que possui somente coreanos e é formado principalmente por pessoas de nível de educação moderado. Segundo os autores, a arquitetura é a primeira baseada em CNN para resolver uma tarefa de classificação de Alzheimer utilizando imagens 2D como entrada. Essa arquitetura é constituída de uma etapa de extração de características usando uma Inception V4 e uma segunda etapa em que essas características são concatenadas com a idade, sexo e número de fatias a serem avaliadas para então serem classificadas por uma camada FC e uma camada *softmax*. Essa arquitetura foi treinada separadamente em cada dos banco de imagens (ADNI e SNUBH) e avaliados em cada uma delas, sendo que, quando treinado e testado no mesmo foi denotado como “dentro do banco de dados” (*within-dataset*) e quando treinado e testado em banco de imagens diferentes foi denotado “entre banco de dados” (*between-dataset*). Otimizado usando um *batch gradient descent* com Nesterov Momentum, o resultado foi muito próximo para o testes *within-dataset* e *between-dataset*, apresentando para as métricas AUC, acurácia, sensitividade e especificidade valores acima de 88%, 83%, 76% e 85%, respectivamente.

Utilizando os dados da ADNI, o trabalho de [101] faz a comparação entre dois algoritmos na classificação de pacientes em 3 classes: saudáveis, MCI e com Alzheimer. O primeiro é uma CNN com 8 camadas convolucionais, 5 camadas de *max-pooling*, 2 camadas FC e 1 camada *softmax*. O segundo é o DML (do inglês, *Deep Metric Learning*) que consiste em um algoritmo capaz de criar uma métrica de distância que aproxima amostras do conjunto de dados pertencentes à mesma classe e afasta as diferentes [101]. O trabalho resultou em uma acurácia, sensitividade e especificidade de 83%, 83% e 82% para o DML contra 81%, 79% e 83% para a CNN. Além disso, mostrou-se que a DML teve uma convergência e desempenho computacional melhor.

Helaly, Badawy e Haikal propõem 3 arquiteturas para a classificação da doença de Alzheimer em [102]. A 2D-M²IC é uma CNN que utiliza 3 camadas convolucionais seguidas de camadas de *max-pooling*, 2 camadas FC e uma camada *softmax* para classificar imagens 2D.

A 3D-M²IC é uma CNN similar a anterior, no entanto ela usa imagens 3D para sua classificação e, portanto, suas camadas convolucionais e de *pooling* possuem uma dimensão a mais. A última arquitetura é uma VGG 19 pré-treinada em que foi acrescentado ao final mais 4 camadas FC e uma *softmax* e foi aplicada a técnica de *fine-tuning*. Imagens do banco de dados ADNI [96] foram balanceadas e ampliadas usando a técnica de aumento de dados para serem utilizadas no treinamento dessas arquiteturas. As arquiteturas obtiveram uma acurácia de 93,60% para a 2D-M²IC, 95,17% para a 3D-M²IC e 97,00% para a VGG 19 pré-treinada para a classificação das classes NC (do inglês, *Normal Control*), EMCI (do inglês, *Early Mild Cognitive Impairment*), LMCI (do inglês, *Late Mild Cognitive Impairment*) e AD (do inglês, *Alzheimer Disease*).

Finalmente, em [103], é proposto uma arquitetura que utiliza dados da ADNI [96] para a classificação entre NC, MCI, EMCI, LMC e AD. Essa arquitetura é uma ResNet pré-treinada em que: a última camada FC foi modificada para realizar a classificação entre uma das 4 classes apresentadas, foi adicionada uma camada de função de ativação com *dropout* e todos seus pesos foram reajustados no processo de *fine-tuning*. Essa ResNet 18 modificada foi avaliada em 7 classificações binárias (NC x AD, NC x EMCI, NC x LMCI, EMCI x LMCI, EMCI x AD, LMCI x AD e MCI x EMCI) com e sem a técnica de *dropout* e, segundo os autores, o modelo foi eficiente no diagnóstico precoce do Alzheimer sem qualquer falso positivo e com muito baixo falso negativo, resultando em uma acurácia mínima de 74,91% e máxima de 99,99% na configuração com a regularização. Além disso, é discutido que se observou o problema de *overfitting*, razão do uso da técnica de *dropout* e *early stopping*³.

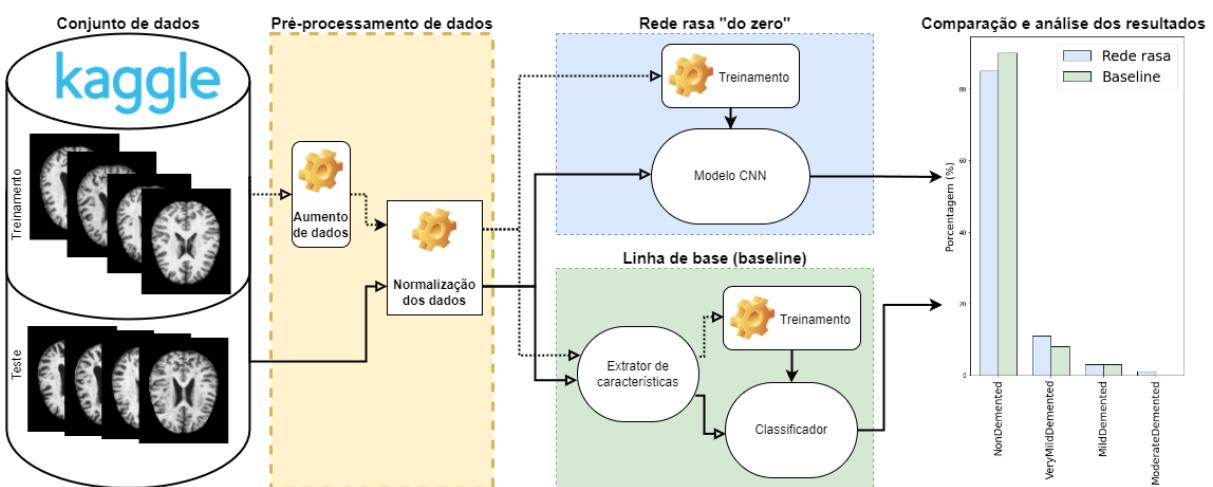
³ *Early stopping* é uma técnica de combate ao problema de *overfitting* que interrompe o treinamento uma vez que o erro de validação atinge seu valor mínimo [104].

3 METODOLOGIA E DESENVOLVIMENTO

3.1 Proposta

A proposta desse trabalho é utilizar os dados rotulados presentes em [29] no treinamento de arquiteturas rasas de CNN projetadas do seu princípio a fim de obter um modelo capaz de classificar uma imagem de MR em: *NonDemented*, *VeryMildDemented*, *MildDemented* ou *ModerateDemented*. Um diagrama da configuração dessa proposta (*pipeline*) é mostrado na Figura 15.

Figura 15 – *Pipeline* da proposta. No lado esquerdo, observa-se que o conjunto de dados divididos em treino e teste, seguido por um bloco de pré-processamento (amarelo). No centro da figura existem dois blocos um de rede rasa (azul), que treina um modelo CNN, e um da linha de base (verde), que treina um classificador utilizando extração de características. Por fim, um bloco de comparação e análise dos resultados.



Fonte: Imagem do autor

Neste *pipeline*, será obtido o conjunto de dados formado por imagens de ressonância magnética e já dividido em treino e teste em [29]. Ambos os subconjunto são pré-processados, sendo que o de treino terá uma etapa de aumento de dados antes da normalização e o de teste somente a normalização. Posterior ao pré-processamento, um modelo de rede rasa e de *baseline* serão treinados utilizando os dados de treinamento e avaliados com o de teste. Finalmente, o resultado obtido do modelo de rede rasa “*from scratch*” (do zero) será comparado ao resultado do *baseline* e uma análise será feita.

Por esse trabalho lidar com um problema de multiclassificação, métricas como acurácia binária, não seriam adequadas, mas, assim como mencionado no item 2.3.3, é possível utilizar

métricas análogas. Por isso será utilizado a acurácia multiclasse que será referido como acurácia global, para diferenciar da acurácia obtida para cada classe. Para reforçar as análises serão apresentadas também tabelas e gráficos.

Para o desenvolvimento do projeto será utilizado python em um sistema operacional Windows 10 utilizado um computador pessoal com a configuração apresentada na Tabela 1 e seu código-fonte estará disponível em ‘github.com/ishibeissao/alzheimer’.

Tabela 1 - Configuração do hardware utilizado na execução do código utilizado no projeto

Hardware	Características
Processador	Intel i5-9600K Coffee Lake Refresh com 9MB de cache e frequência de 3.7GHz
Placa gráfica	NVIDIA GeForce GTX 1660 SUPER SC Ultra Gaming com 6GB de memória dedicada
Memória RAM	2 pentes DDR4 de 8GB com frequência de 3000MHz
SSD	Tipo M.2 NVMe, com taxa de leituras e gravação de 1700Mb/s e 1550Mb/s, respectivamente

Fonte: Tabela do autor

Cada um dos blocos mencionados e apresentados na Figura 15 (conjunto de dados, pré-processamento, linha de base, rede rasa “do zero” e comparação e análise dos resultados) serão melhor explicados nos itens seguintes.

3.2 Conjunto de dados

O conjunto de imagens de ressonância magnética obtido de [29] consistem em fatias horizontais de diferentes posições do cérebro⁴. Todas estão na escala de cinza, possuem 208 pixels de altura e 176 pixels de largura e estão alinhadas e centralizadas, como mostra a Tabela 2.

⁴ Em [105], é possível observar o formato e posição de diferentes fatias horizontais, bem como o nome das estruturas.

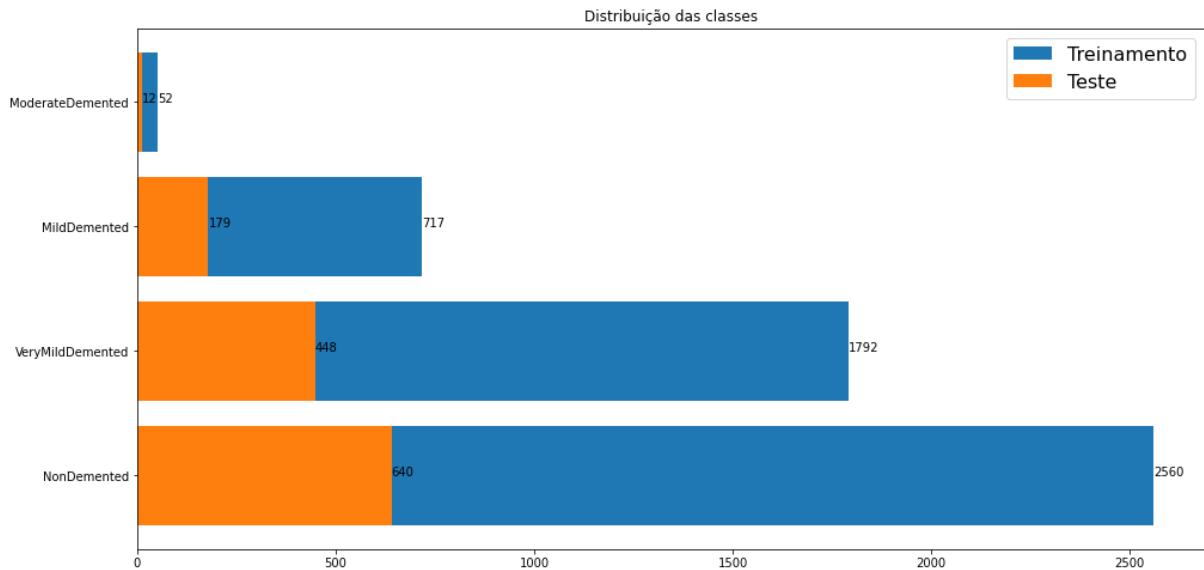
Tabela 2 – Exemplos de imagens de ressonância magnética para cada uma das 4 classes: NonDemented, VeryMildDemented, MildDemented e ModerateDemented presentes em [29]. As imagens podem ser provenientes de fatias horizontais de diferentes posições do cérebro

	Exemplo 1	Exemplo 2	Exemplo 3	Exemplo 4	Exemplo 1
NonDemented					
VeryMildDemented					
MildDemented					
ModerateDemented					

Fonte: Tabela do autor

Esse conjunto de dados é formado por 6400 imagens de ressonância magnética do cérebro, divididos em 5121 de treino e 1279 de teste, representando assim, 80% e 20% do total. As imagens de treino e teste são distribuídas entre as classes conforme a Figura 16.

Figura 16 – Distribuição das classes no conjunto de treino (5121 imagens) e no conjunto de teste (1279 imagens)



Fonte: Imagem do autor

Com base na Figura 16, pode se observar que as classes não são igualmente distribuídas, sendo que a classe *NonDemented* representa a metade de todo o conjunto de imagens. Essa característica do conjunto de dados pode prejudicar o treinamento de uma CNN, pois as classes minoritárias podem acabar sendo ignoradas devido a sua baixa representatividade no conjunto total.

Observa-se também que a separação entre treino e teste segue a mesma distribuição, o que indica uma divisão apropriada dessas imagens.

Outra característica que deve ser mencionada é a dificuldade que as diferentes fatias do cérebro impõem sobre o projeto, uma vez que elas possuem formatos diferentes e não há nenhuma indicação no conjunto de dados que identifica em qual posição do cérebro a imagem foi coletada. Além do mais, essa característica aumenta ainda mais a diversidade de imagens, que pode sugerir que o tamanho do conjunto de dados não seja grande o suficiente.

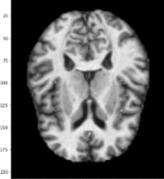
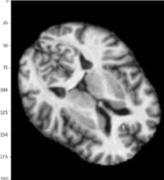
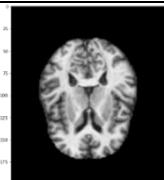
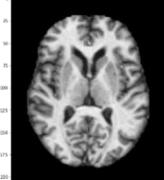
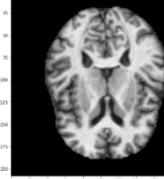
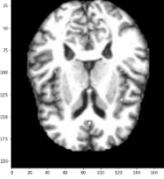
3.3 Pré-processamento

Neste trabalho, o pré-processamento consiste em aplicar as técnicas de aumento de dados e normalização no subconjunto de treino e somente a normalização no subconjunto de teste.

Embora se possa aplicar diferentes pesos durante a fase de treinamento, privilegiando as classes minoritárias, para reduzir o problema do desbalanceamento, o aumento de dados também traz o benefício de reduzir a susceptibilidade dos modelos a distorções na imagem.

Para o aumento de dados serão utilizadas as operações de transformação nas imagem de treinamento apresentadas na Tabela 3.

Tabela 3 - Operações de transformação de imagem utilizadas no aumento de dados. Na tabela também é apresentado a variação permitida para cada operação e um exemplo de transformação

Operação	Parâmetros da operação	Descrição do exemplo	Exemplo de imagem
Sem operação	Não há	Imagen original	
Rotação	Rotações permitidas: 45°, 90°, 135°, 180°, 225°, 270° e 315°	Giro de 45° no sentido anti-horário	
Redimensionamento	Redução ou aumento de até 20% do tamanho original	Redução de 20% do tamanho original	
Inversão de eixo	Inversão do eixo x, y ou de ambos eixos	Inversão do eixo y	
Deslocamento	Deslocamento de 20 pixels tanto na vertical como na horizontal em ambas direções	Deslocamento em 10 pixels para a direita	
Alteração de intensidade	Aumento ou redução da intensidade do pixel em até 20%	Aumento de 20% do valor do pixel	

Fonte: Tabela do autor

Também na Tabela 3, foram definidas as variações possíveis de cada operação de transformação, por exemplo, foi definido que pode ser aplicado um deslocamento em até 20 pixel tanto na vertical como na horizontal e em ambos sentidos. Esses valores foram definidos pela observação das imagens do conjunto de dados

Tais operações também podem ser combinadas, aplicando uma após a outra de modo a obter uma nova transformação, por exemplo, aplicar uma rotação e um redimensionamento.

Considerando que se pode realizar 5 tipos de operações e elas podem ser combinadas, sem incluir repetições ou ordem das operações, temos a Tabela 4.

Tabela 4 - Número de transformações possíveis para 5 tipos de operações, sem considerar repetições ou ordem das operações. Na coluna combinações, separado por vírgulas estão todas as transformações possíveis e entre colchetes está a combinação de operações

Sequência de operações	Número de transformações possíveis	Combinações
1	5	[Rt], [Rd], [I], [D], [A]
2	10	[Rt, Rd], [Rt, I], [Rt, D], [Rt, A], [Rd, I], [Rd, D], [Rd, A], [I, D], [I, A], [D, A]
3	10	[Rt, Rd, I], [Rt, Rd, D], [Rt, Rd, A], [Rt, I, D], [Rt, I, A], [Rt, D, A], [Rd, I, D], [Rd, I, A], [Rd, D, A], [I, D, A]
4	5	[Rt, Rd, I, D], [Rt, Rd, I, A], [Rt, Rd, D, A], [Rt, I, D, A], [Rd, I, D, A]
5	1	[Rt, Rd, I, D, A]
Total	31	-

Rt: rotação, Rd: redimensionamento, I: inversão de eixo, D: deslocamento e A: alteração de brilho

Fonte: Tabela do autor

Assim, o aumento de dados consiste em aplicar aleatoriamente uma dessas 31 operações de transformações em imagens do subconjunto de treino e adicioná-la ao próprio subconjunto.

A quantidade de novas imagens a serem adicionadas deve ser de tal forma que:

- Haja balanceamento das classes; e
- Todas as classes tenham uma porção de imagens transformadas.

Desta maneira, não só se combate o problema da distribuição das classes, mencionado no tópico 3.2, mas também garante-se um nível de robustez a variações qualquer que seja a imagem.

O número de imagens a serem produzidas pela técnica de aumento de dados é apresentado na Tabela 5.

Tabela 5 – Comparaçao entre a quantidade esperada de dados antes e depois do aumento de dados

Classe	Tamanho original	Imagens aumentadas	Fator de aumento de dados	Tamanho final (originais + transformadas)	Porcentagem do total final (%)
Non Demented	2560	2816	1,1	5376	25,14
Very Mild Demented	1792	3584	2	5376	25,14
Mild Demented	717	4660	6,5	5377	25,15
Moderate Demented	52	5200	100	5252	24,57
Total	5121	16260	4,175	21381	100

Fonte: Tabela do autor

Finalmente, a normalização das imagens, que é aplicada no subconjunto de treino aumentado e no subconjunto de teste, consiste no reescalonamento do valor dos pixel da imagem que varia de 0 a 255 para um valor entre 0 e 1. Para tanto, basta dividir todos os valores obtidos da imagem por 255.

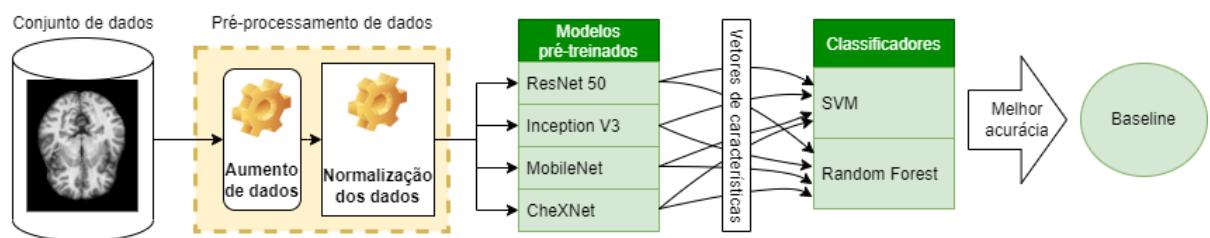
3.4 Linha de base (*baseline*)

A linha de base ou *baseline* é uma referêcia proposta a fim de avaliar o resultado de um projeto. Diversas são as maneiras de se definir um *baseline*, uma forma simples seria considerar a probabilidade da classe correta ser escolhida ao acaso, assim, para um problema

de N classes o *baseline* seria $1/N$. Outra maneira seria usar um resultado apresentado na literatura como referência.

Assim como mostrado na Figura 15, neste projeto será utilizado como *baseline* o resultado obtido com a extração de características de um modelo pré-treinado e classificado em um classificador de convencional, adotado regularmente em estudos de reconhecimento de padrões. A escolha desse modelo pré-treinado e classificador será a combinação que apresentar a melhor acurácia para o conjunto de dados utilizados. A Figura 17 ilustra o processo de extração de características, bem como os modelos e classificadores utilizados.

Figura 17 – Diagrama do processo de extração de características. O conjunto de dados pré-processado é transformado em vetores de características através dos modelos pré-treinados (extratores de características). Cada um dos 4 vetores de características gerado é dado como entrada para os classificadores. A combinação que apresentar melhor acurácia no conjunto de teste será utilizada como *baseline*



Fonte: Imagem do autor

As escolhas dos modelos pré-treinados se deram da seguinte maneira:

- O ResNet 50 e Inception V3 foram escolhidos devido a seus resultados na classificação do ImageNet, além de serem utilizados em muitos outros trabalhos da literatura.
- O MobileNet possui a característica de ser um modelo eficiente em aplicações que possuem restrições de recursos computacionais como dispositivos embarcados [86].
- O CheXNet é um modelo treinado para detecção de pneumonia em imagens de raio-X do tórax, e, devido a isso, possui maior similaridade ao conjunto de dados médico utilizado. [89]

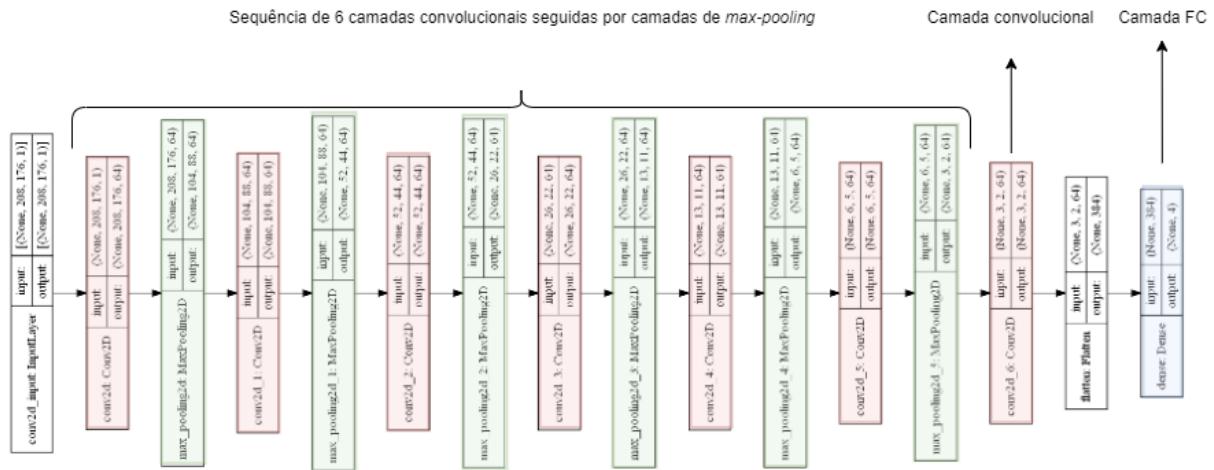
Os classificadores de baixo viés escolhidos foram 2 popularmente utilizados: SVM (do inglês, *Support Vector Machine*) e *Random Forest*.

A combinação desses modelos pré-treinados e classificadores produzirá 8 resultados. O *baseline* será adotado como o resultado que obtiver maior acurácia sob o conjunto de teste.

3.5 Rede rasa “do zero”

A rede rasa “do zero” trata-se de uma CNN criada do seu princípio, sem a utilização de técnicas de transferência de aprendizado ou incorporações de partes ou estruturas de outros modelos como blocos *inception*, *skipping layers* ou *dense blocks*. Assim, algumas redes (arquiteturas) são propostas combinando uma sequência de camadas convolucionais seguidas camadas de *pooling*, uma camada convolucionais e uma FC. A Figura 18 mostra uma representação gerada em Python usando a biblioteca Keras para uma rede que possui 6 pares de camadas convolucionais e *max-pooling* (6 pares convolucionais/*max-pooling*), seguida de uma camada convolucional e uma FC, que foi um dos resultados desse trabalho.

Figura 18 – Representação da rede rasa F formada por 7 camadas convolucionais (vermelho), 6 camadas *max-pooling* (verde) e 1 camada FC (azul).



Fonte: Imagem do autor

Nessa rede, as camadas convolucionais possuem 64 filtros de 3x3, *stride* de 1 em ambas dimensões, *pad* de 1 pixel em toda a borda e função de ativação do tipo ReLu. As camadas de *max-pooling* possuem uma janela de 2x2 com *stride* de 2 em ambas direções. A técnica de *dropout* pode ser empregada, caso seja necessária, com uma taxa entre 10% e 50%. E, camada de FC com função de *softmax* para 4 classes. Variando-se o números de pares convolucionais/*max-pooling* de 1 a 7 são propostas as redes A a G mostradas na Tabela 6.

Tabela 6 – Redes rasas “do zero” propostas. São formadas por uma sequência de camadas convolucionais seguidas por uma de *max-pooling* e, ao fim, uma camada convolucional e uma FC

Nome	Sequência de camadas convolucionais e <i>max-pooling</i>	Número de parâmetros
A	1	2380484
B	2	660228
C	3	257860
D	4	184964
E	5	192964
F	6	223748
G	7	259396

Fonte: Tabela do autor

A quantidade de filtros das camadas convolucionais foram convenientemente fixadas em 64 filtros de acordo com algumas medidas preliminares para garantir um bom resultado sem o aumento demasiado do tempo de processamento, considerando as configurações de *hardware* apresentadas na Tabela 1.

O tamanho dos filtros de 3x3 foi mantido, pois, assim como dito pelos autores da rede VGG, isso torna a função de decisão mais discriminativa.

A cada camada de *max-pooling* de janela de 2x2, a imagem de entrada reduz sua altura e largura pela metade. A Tabela 7 mostra o tamanho da saída da última camada *max-pooling* para cada uma das redes rasas.

Tabela 7 – Dimensão do vetor de características após a última camada *max-pooling* para cada rede rasa.

Rede rasa	A	B	C	D	E	F	G
Pixels	104 x 88	52 x 44	26 x 22	13 x 11	6 x 5	3 x 2	1 x 1

Fonte: Tabela do autor

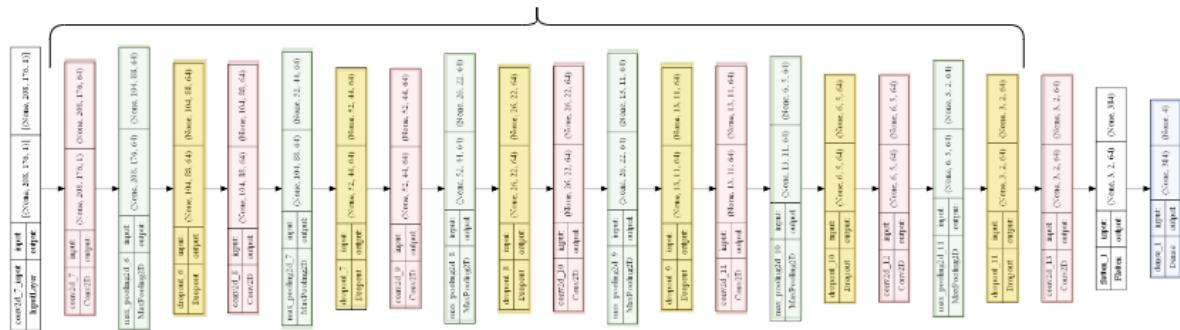
Devido a isso, uma imagem com 208 pixels de altura e 176 pixels de largura permite que somente 7 camadas *max-pooling* sejam colocadas em sequência na rede, pois, ao fim da

séptima camada, o vetor de características terá dimensão mínima de 1 x 1. Por isso, será utilizado somente 7 configurações (A a G).

Quando utilizado, o *dropout* é aplicado após cada par convolucionais/*max-pooling* assim como representado na Figura 19. Essa figura é a rede rasa F com 15% de taxa de *dropout* que resultou esse trabalho.

Figura 19 - Representação da rede rasa F com *dropout* de 15% formada por 7 camadas convolucionais (vermelho), 6 camadas *max-pooling* (verde) e 1 camada FC (azul). Em amarelo, estão as camadas de *dropout* do Keras e são aplicadas após cada par convolucionais/*max-pooling*

Sequência de 6 camadas convolucionais seguidas por camadas de *max-pooling* com *dropout* de 15%



Fonte: Imagem do autor

Essas 7 redes rasas serão treinadas usando o otimizador Adam com as configurações padrão do Keras, *batch size* de 64 em 100 épocas. E, após treinadas, serão comparadas uma as outras e aquela que produzir a que obtiver maior acurácia será utilizada.

3.6 Comparação e análise dos resultados

O bloco final do *pipeline* consiste em analisar os resultados da rede rasa do “zero” com base no *baseline*. Essa análise será feita por meio da medida da acurácia multiclasse, ainda apontando os sucessos, dificuldades e escolhas (*trade-offs*).

4 RESULTADOS E ANÁLISE

Os resultados obtidos de cada um dos blocos do *pipeline* são apresentados abaixo, bem como uma análise acerca deles.

4.1 Pré-processamento de dados

Assim, a execução do aumento de dados produziu os resultados mostrados nas Tabela 8 e Tabela 9.

Tabela 8 – Número de operações realizadas por imagem para cada uma das classes. O subtotal A indica a quantidade de imagens geradas pelo aumento de dados por classe e o subtotal B a quantidade de imagens produzidas por número de operações

Número de operações	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Subtotal B
1	567	713	884	1034	3198 (19,67%)
2	540	701	934	1043	3218 (19,79%)
3	588	705	928	1040	3261 (20,06%)
4	565	785	900	1025	3275 (20,14%)
5	556	680	1014	1058	3308 (20,34%)
Subtotal A	2816	3584	4660	5200	16260

Fonte: Tabela do autor

Na tabela anterior é apresentado o número de imagens em que foram aplicada de 1 a 5 operações de transformação por classe. Por exemplo, o dado presente na terceira linha e primeira coluna indica que 588 imagens da classe *Non demented* foram produzidas utilizando 3 operações de transformação. É possível observar que o número de imagens geradas pela técnica de aumento de dados por classe, indicado pelo subtotal A, apresenta os mesmos valores da coluna “Imagens aumentadas” da Tabela 5 e o número de imagens produzidas por número de operações de transformações, indicado pelo subtotal B, é bem distribuído.

Tabela 9 – Quantidade de operações de transformação realizada pela técnica de aumento de dados por classe

Operação	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented
Alteração de intensidade	1710	2123	2834	3140
Inversão de eixo	1698	2172	2815	3151
Deslocamento	1690	2148	2830	3122
Redimensionamento	1706	2173	2871	3127
Rotação	1647	2154	2856	3090

Fonte: Tabela do autor

Na Tabela 9, é mostrada a quantidade de operações de transformações utilizadas para produção das 16260 imagens do aumento de dados. Nesta tabela, os valores indicam o número total de imagens que foram transformadas pela operação independentemente do número de operações realizadas, portanto, uma mesma imagem transformada usando deslocamento e rotação será contabilizada na linha de ambas operações. Os valores mostram que foram aplicados o mesmo número de operações para produção das imagens do aumento de dados, não havendo tendência para uma das transformações.

Assim, os resultados da Tabela 8 e Tabela 9 sugerem que a técnica de aumento de dados produziu os resultados esperados no item 3.3, com o fim de combater o desbalanceamento dos dados e conferir robustez a variações das imagens, sem introduzir tendência ou padrões ao novo conjunto de treino que poderiam reduzir o desempenho dos modelos treinados.

4.2 Linha de base (*baseline*)

Assim como mostrado na Figura 17, cada um dos modelos pré-treinados gerou um vetor de características a partir das 21381 imagens pré-processadas (4.1). Esses vetores de características foram utilizados no treinamento dos classificadores SVM e *Random Forest*. A

Tabela 10 e Tabela 11 apresentam, respectivamente, o resultado para o SVM e o *Random Forest* para cada classe avaliados no conjunto de teste.

Tabela 10 – Acurácia obtida do conjunto de teste avaliado no classificador SVM utilizando vetores de características extraídos por diferentes modelos pré-treinados. As 4 primeiras colunas mostram a acurácia para cada uma das classes e a última, uma acurácia global

Modelo pré-treinado	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Global
ResNet 50	69,06	29,02	32,96	50,00	49,80
Inception V3	51,41	30,80	30,73	33,33	41,13
MobileNet	59,84	26,34	19,55	33,33	42,22
CheXNet	51,09	28,79	38,55	91,67	41,91

Fonte: Tabela do autor

Tabela 11 - Acurácia obtida do conjunto de teste avaliado no classificador *Random Forest* utilizando vetores de características extraídos por diferentes modelos pré-treinados. As 4 primeiras colunas mostram a acurácia para cada uma das classes e a última, uma acurácia global

Modelo pré-treinado	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Global
ResNet 50	63,12	41,52	16,20	8,33	48,48
Inception V3	70,47	29,91	23,46	0,00	49,02
MobileNet	43,12	21,43	17,88	16,67	31,74
CheXNet	32,34	66,74	31,84	8,33	44,10

Fonte: Tabela do autor

Pelos resultados, observa-se que metade dos modelos pré-treinados apresentaram melhor desempenho com o SVM e metade com o *Random Forest*, sendo que o primeiro foi ligeiramente melhor com média de acurácia global de 43,77% contra 43,34%, além de obter a melhor acurácia com a ResNet50 de 49,80%.

Outro ponto a ser observado é o padrão de acurácia das classes, enquanto que o SVM possui maiores taxas de acerto para as os estágios extremos da doença (*Non Demented* e *Moderate Demented*), o *Random Forest* apresenta uma queda conforme os estágios avançam.

Se considerarmos o valor mínimo de acurácia entre as classes para ambos os classificadores, assim como mostrado na Tabela 12, observamos que o *Random Forest* sempre possui o menor valor, chegando a 0% para o Inception V3.

Tabela 12 – Valor mínimo de acurácia entre as classes para cada classificador e modelo pré-treinado. Juntamente com o valor mínimo está a classe associada

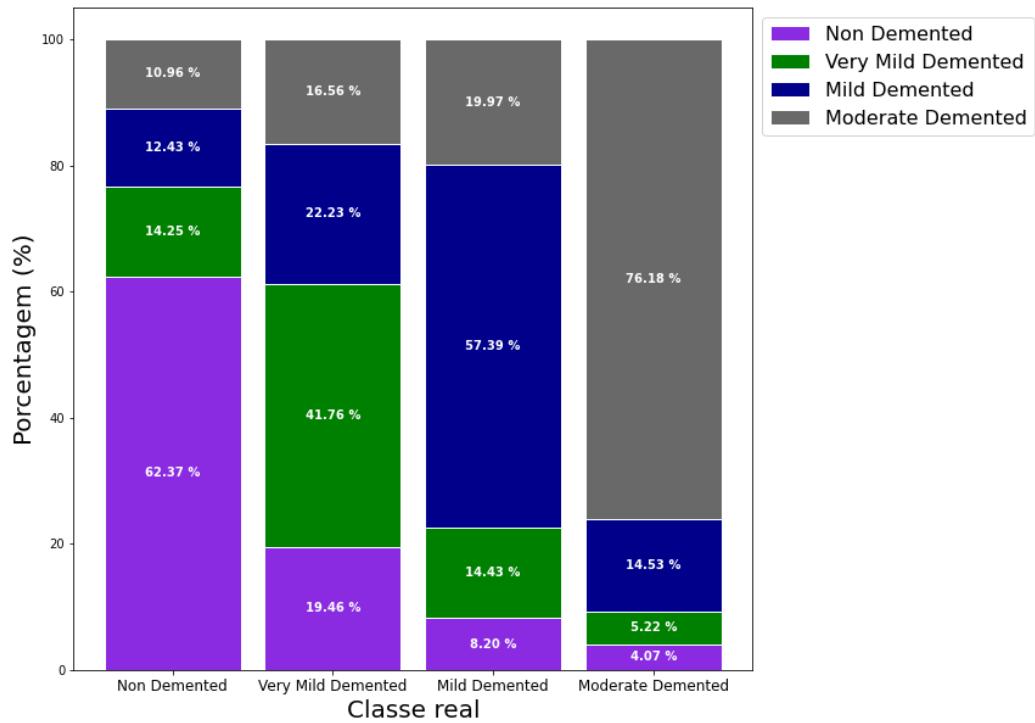
Modelo pré-treinado	SVM	Random Forest
ResNet 50	29,02 (Very Mild Demented)	8,33 (Moderate Demented)
Inception V3	30,73 (Mild Demented)	0,00 (Moderate Demented)
MobileNet	19,55 (Mild Demented)	16,67 (Moderate Demented)
CheXNet	28,79 (Very Mild Demented)	8,33 (Moderate Demented)

Fonte: Tabela do autor

Embora a acurácia global seja uma métrica que facilite a comparação entre modelos, pois representa a taxa de acerto geral, é importante que seja verificado as acuráncias por classes para que garantir que nenhuma classe está sendo desfavorecida. Por conseguinte, foi decidido optar pela combinação de modelo pré-treinado e classificador ResNet 50 e SVM com 49,80%, não só por ter resultado na maior acurácia global, mas também por possuir um equilíbrio entre a acurácia das classes.

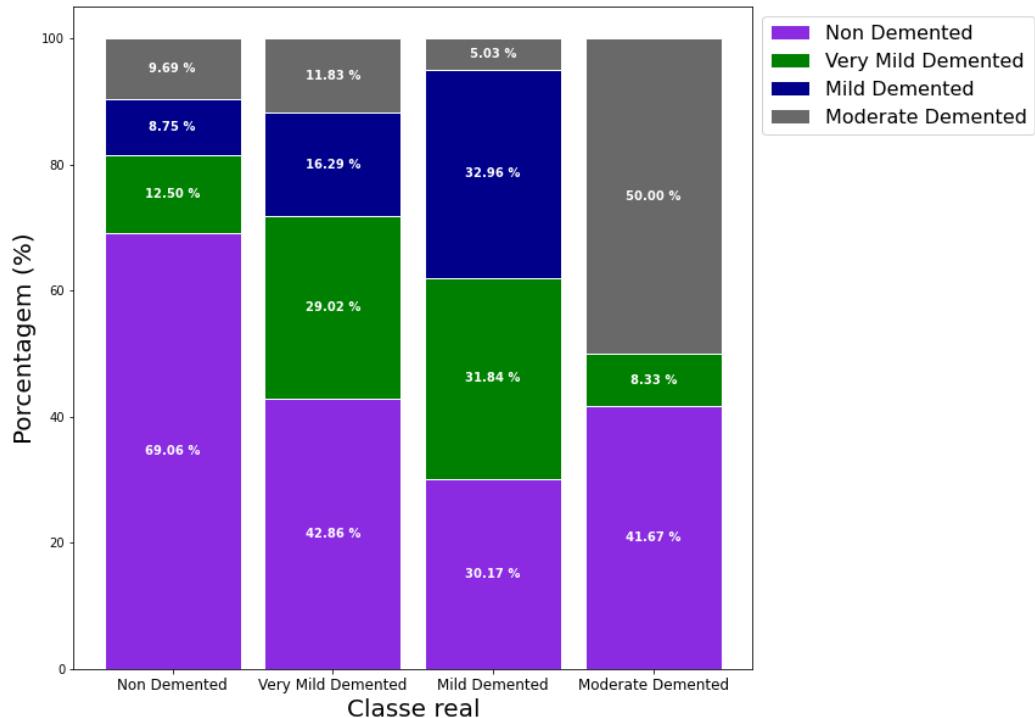
A fim de fazer uma melhor análise do *baseline* são mostrados gráficos apresentando a porcentagem de classe predita para cada classe real tanto do conjunto de treino (Figura 20) como do conjunto de teste (Figura 21).

Figura 20 - Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) do modelo *baseline* (ResNet 50 e SVM) avaliado no conjunto de treino



Fonte: Imagem do autor

Figura 21 – Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) do modelo *baseline* (ResNet 50 e SVM) avaliado no conjunto de teste



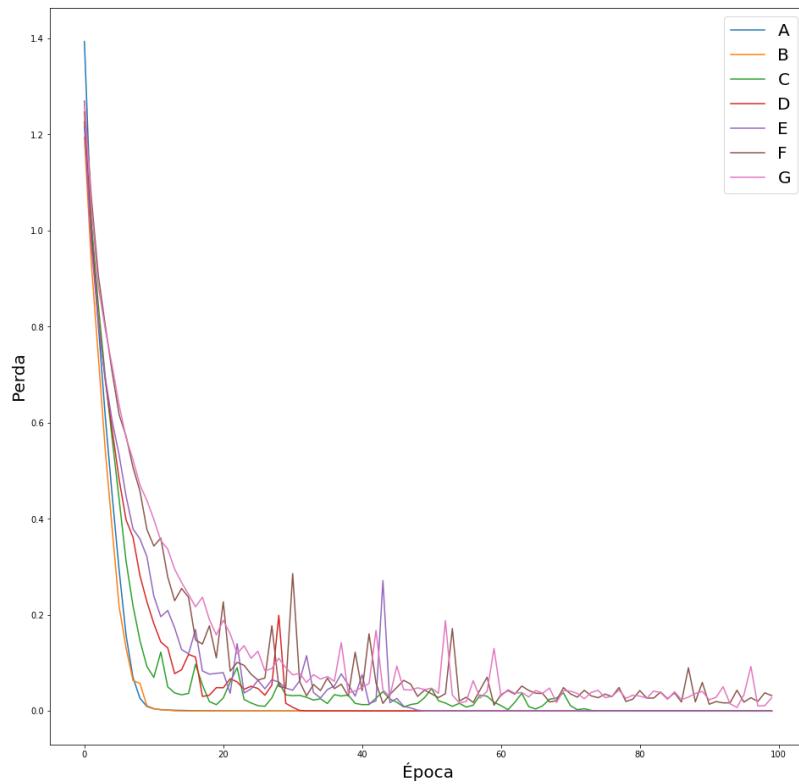
Fonte: Imagem do autor

Os resultados de acurácia do *baseline* foram acima de 25%, portanto, maior que uma seleção aleatória. Entretanto, evidencia-se também a dificuldade imposta pelo conjunto de dados na separação entre as classes intermediárias *Very Mild Demented* e *Mild Demented*. A primeira possui uma acurácia inferior a 50% no conjunto de treinamento e não é a classe majoritária no conjunto de teste, podendo levar a diversos diagnósticos falso-negativos para a doença. A segunda não atinge 50% de acurácia no conjunto de teste e ainda pode levar a quase 30% de resultados falso-negativos para a doença. Isso reforça a dificuldade existente de detecção da doença.

4.3 Rede rasa “do zero”

É mostrado na Figura 22 o gráfico da perda por época para cada uma das redes rasas propostas e nas Tabela 13 e Tabela 14 os resultados de acurácia para os conjuntos de treino e teste para cada uma delas.

Figura 22 – Gráfico da perda por época durante o treinamento das redes rasas sem *dropout*



Fonte: Imagem do autor

Tabela 13 - Acurácia obtida do conjunto de treino para cada rede rasa proposta. As 4 primeiras colunas mostram a acurácia para cada um das classes e a última, uma acurácia global. Para o treinamento foram utilizadas as imagens com aumento de dados e regularização e não foi aplicado a técnica de *dropout*

Rede rasa	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Global
A	100,00	100,00	100,00	100,00	100,00
B	100,00	100,00	100,00	100,00	100,00
C	100,00	100,00	100,00	100,00	100,00
D	100,00	100,00	100,00	100,00	100,00
E	100,00	100,00	100,00	100,00	100,00
F	99,96	97,17	98,33	99,89	98,83
G	99,70	98,75	99,52	99,98	99,49

Fonte: Tabela do autor

Tabela 14 – Acurácia obtida do conjunto de teste para cada rede rasa proposta. As 4 primeiras colunas mostram a acurácia para cada um das classes e a última, uma acurácia global. Para o treinamento foram utilizadas as imagens com aumento de dados e regularização e não foi aplicado a técnica de *dropout*

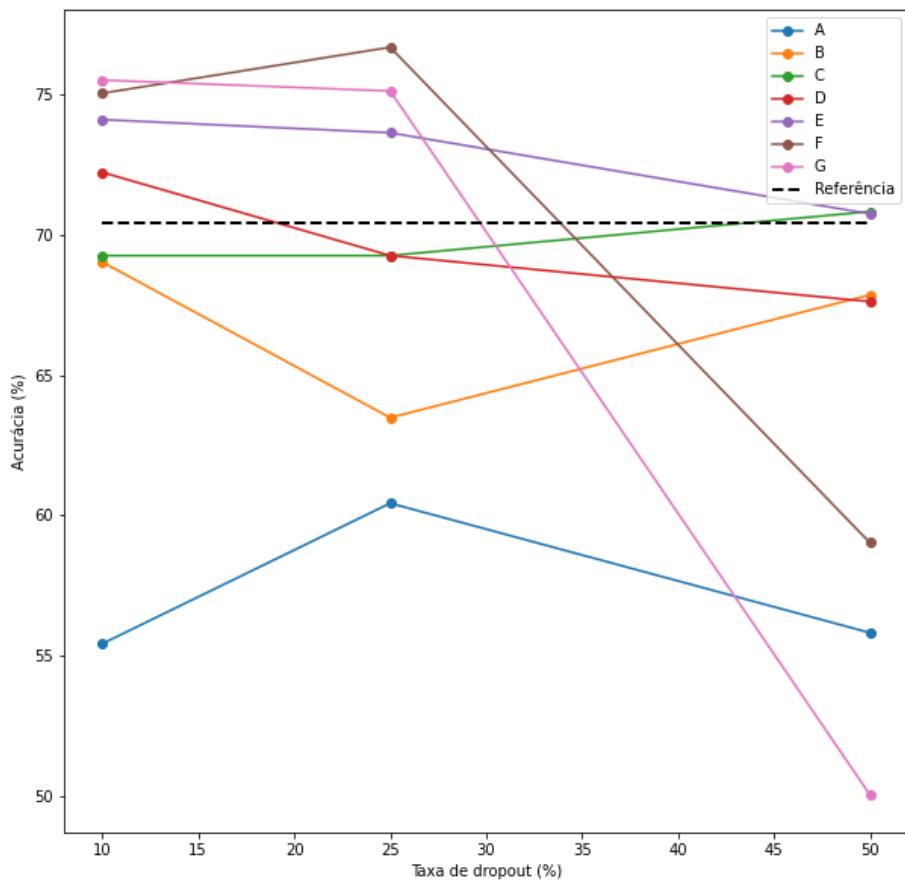
Rede rasa	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Global
A	71,41	43,97	53,07	75,00	59,27
B	80,62	51,56	50,84	50,00	65,99
C	72,50	62,05	49,16	41,67	65,29
D	73,75	76,12	39,11	58,33	69,59
E	80,16	57,14	63,69	75,00	69,74
F	86,09	53,35	56,98	75,00	70,45
G	83,75	52,01	62,57	25,00	69,12

Fonte: Tabela do autor

O gráfico de perda indica que houve convergência durante o treinamento e o resultado obtido mostra uma alta acurácia para no conjunto de treinamento em relação ao conjunto de teste, indicando que possivelmente há um problema de *overfitting*.

Assim, foi implementado a técnica de *dropout* com taxas de 10%, 25% e 50% nas diferentes redes. Os resultados estão apresentados na Figura 23.

Figura 23 – Gráfico que mostra a acurácia global pela taxa de *dropout* para as redes rasas propostas usando o conjunto de teste. Em preto, a melhor acurácia obtida entre as redes rasas sem *dropout* (rede rasa F) no conjunto de teste



Fonte: Imagem do autor

Observou-se que há uma melhora na acurácia de algumas redes quando é empregada a técnica de *dropout*. Nota-se também que essa técnica prejudica as redes com menos camadas, pois nessas redes a desativação de um neurônio afeta demasiadamente a sua capacidade de classificação.

Os valores de acurácia apresentados apontam que a rede F com *dropout* de 25% é aquela com o melhor desempenho. No entanto, devido ao número de valores de taxa de *dropout* avaliados e a curva da Figura 23 é de se supor que haja uma taxa que maximize a acurácia da rede F. Por isso, é feito uma varredura por diferentes taxas de *dropout* para a rede F. A Tabela 15 mostra a acurácia global da rede F com a taxa de *dropout* variando de 10% a 40% com passos de 5%.

Tabela 15 – Acurácia global do conjunto de treino e de teste da rede F para diferentes taxas de *dropout*

Taxa de <i>dropout</i> (%)	Acurácia Treino (%)	Acurácia Teste (%)
10	99,69	75,06
15	99,49	78,50
20	98,42	76,08
25	96,02	76,70
30	89,78	74,20
35	87,12	71,38
40	81,65	67,94

Fonte: Tabela do autor

A melhor acurácia obtida foi com a taxa de *dropout* de 15%. A fim de melhor apresentar essa rede é apresentado na Tabela 16, não só a acurácia global, mas também a acurácia por classes obtida do conjunto de treinamento e de teste.

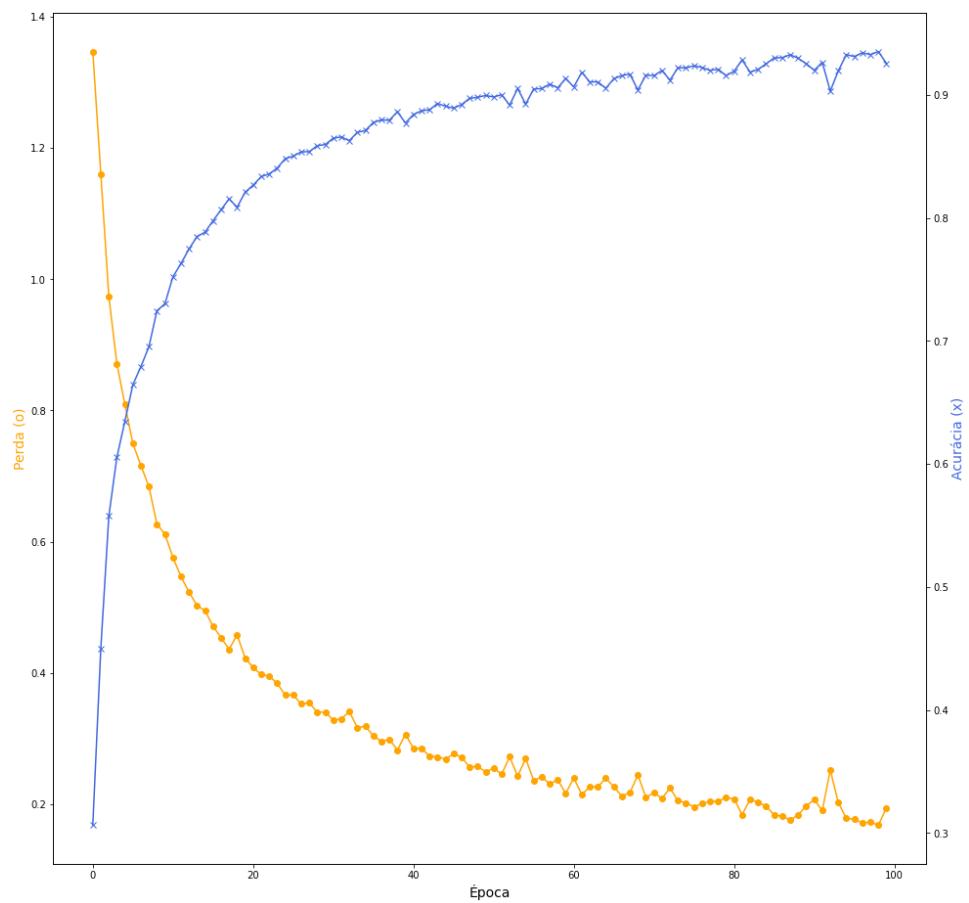
Tabela 16- Acurácia da rede F com *dropout* de 15% avaliado no conjunto de teste e de treino. As 4 primeiras colunas mostram a acurácia para cada uma das classes e a última, uma acurácia global

	Non Demented	Very Mild Demented	Mild Demented	Moderate Demented	Global
Treino	99,76	98,96	99,24	100,00	99,49
Teste	89,06	66,96	69,83	75,00	78,50

Fonte: Tabela do autor

Com isso, são apresentadas as curvas de perda e de acurácia em função da época obtidas durante o treinamento na Figura 24.

Figura 24 – Gráfico da perda (esquerda, laranja) e acurácia (direita, azul) em função da época durante o treinamento da rede F com *dropout* de 15%

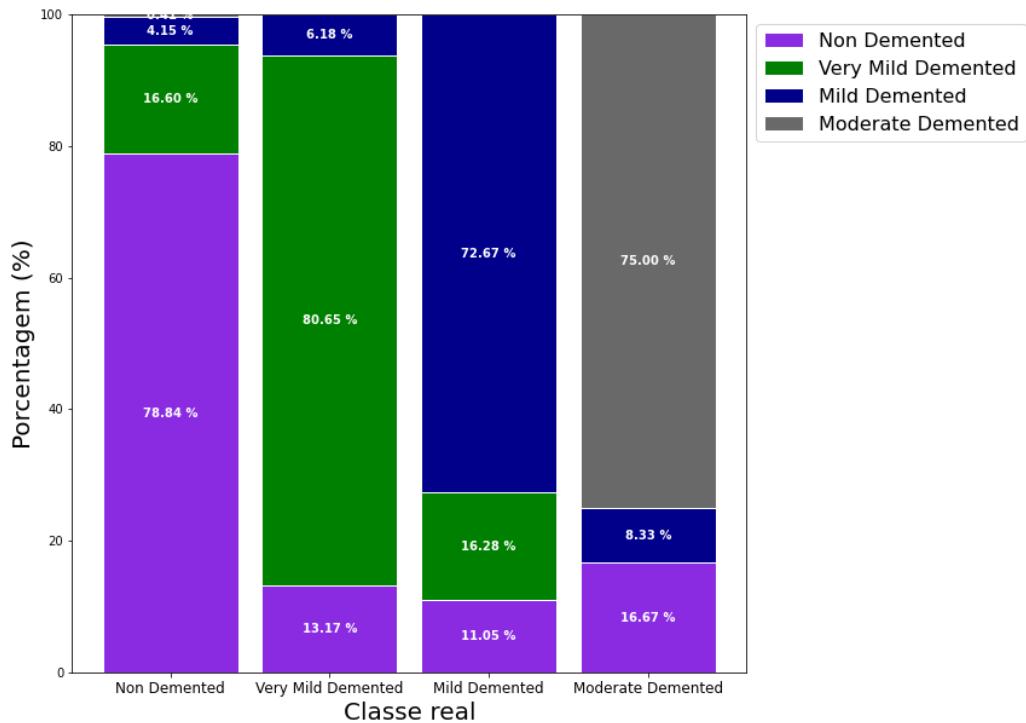


Fonte: Imagem do autor

Pode-se observar que houve uma queda na curva de perda da mesma maneira em que houve um aumento na acurácia, tal como esperado em um treinamento sem problemas de convergência.

Similarmente a rede rasa F sem *dropout*, a acurácia do conjunto de treino foi acima de 98%. Para o conjunto de teste são apresentadas as porcentagens de classe predita para cada classe real na Figura 25.

Figura 25 - Porcentagem predita de cada classe (legenda) para cada classe real (eixo x) da rede F com *dropout* de 15% avaliado no conjunto de teste



Fonte: Imagem do autor

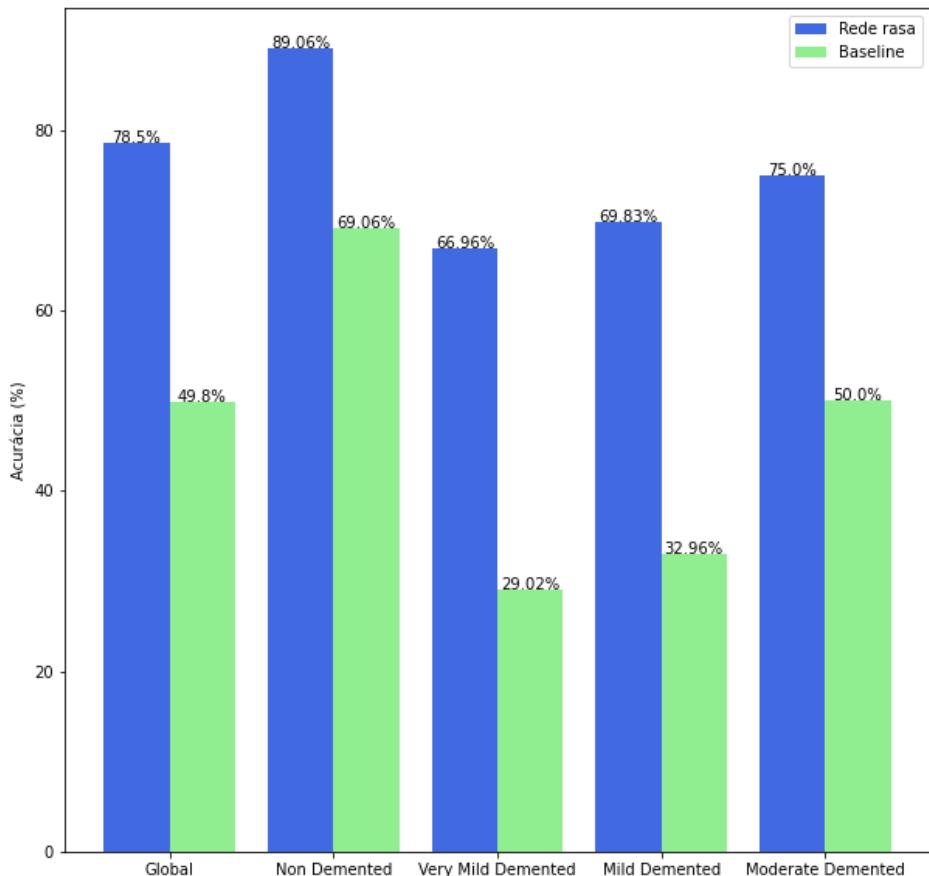
Através de todos esses resultados, é possível dizer que a técnica de *dropout* aumentou a acurácia de rede rasa F em 8,05%. Nesta rede rasa escolhida acurácia obtida do conjunto de treinamento é 21% maior do que no de teste, o que pode indicar ainda certo grau de *overfitting*.

Outro ponto observado é a maior dificuldade da rede de classificar as classes intermediárias (*Very Mild Demented* e *Mild Demented*).

4.4 Comparação e análise dos resultados

Tendo sido treinados e avaliados com o mesmo conjunto de dados pré-processados, o modelo *baseline* e de rede rasa tem seus resultados comparados na Figura 26.

Figura 26 - Comparação entre acurácia global e de cada classe do modelo *baseline* e da rede rasa F com dropout de 15%



Fonte: Imagem do autor

Esse resultado mostra que não só a rede rasa obteve maior acurácia global que o *baseline*, mas também a supera em cada uma das classes. Logo, foi possível projetar uma rede rasa criada do seu princípio com resultado superior à referência estabelecida.

Tal como evidenciado anteriormente, as classes intermediárias da doença (*Very Mild Demented* e *Mild Demented*) possuem uma menor acurácia. Com o propósito de identificar as causas dessa dificuldade será feito um treinamento de uma rede rasa similar a rede escolhida, mas para um problema de classificação binária. Tal rede, que será denotada “rede rasa binária”, terá todas as camadas da rede rasa F (sem *dropout*) com exceção da última que será substituída por camada de *softmax* com somente 2 neurônios. O seu treinamento será realizado com o mesmo conjunto de dados apresentado em 3.2 - Conjunto de dados, mas somente com 2 classes e com os mesmos parâmetros tal como descrito em 3.5 - Rede rasa “do zero”. O resultado é mostrado na Tabela 17.

Tabela 17 - Acurácia da rede rasa F modificada para um problema de classificação binária. A avaliação foi realizada sobre o conjunto de testes para todas as combinações de classes

	Very Mild Demented	Mild Demented	Moderate Demented
Non Demented	74,54	86,81	95,29
Very Mild Demented	-	80,22	97,17
Mild Demented	-	-	93,72

Fonte: Tabela do autor

A classificação binária das classes *Non Demented* x *Very Mild Demented* e *Very Mild Demented* x *Mild Demented* apresentaram as mesmas dificuldades observadas com rede rasa F com *dropout* de 15% da classificação multiclass, indicando que é um obstáculo inerente do problema. Essa análise vai de encontro com a mesma dificuldade existente quando médicos tentam realizar a leitura das imagens de MR na detecção da doença.

Finalmente, os testes foram conduzidos de forma que o tempo de execução do treinamento de cada uma das redes (usando um computador com a configuração da Tabela 1) fosse o suficiente para a viabilidade do projeto considerando prazos, metas e qualidade do resultado.

5 CONCLUSÕES

Através do estudo de arquiteturas e técnicas na literatura combinado com diversos testes e análises foi possível obter uma rede rasa capaz de atingir uma acurácia de 78,5% na classificação de imagens de ressonância magnética do cérebro em: sadio ou em um dos 3 estágios da doença de Alzheimer.

O aumento de dados não só proporcionou uma maior robustez da rede, mas também mitigou o problema do desbalanceamento de classes que existia no conjunto de treinamento. Um indicativo desse último resultado foi sua capacidade de obter um bom valor de acurácia para todas as classes e não se ajustando somente para a classe majoritária (*Non Demented*).

A utilização do *dropout* propiciou um aumento de 8,05% da acurácia da rede rasa F, já que esta apresentava sinais de *overfitting* (alta taxa de acerto no conjunto de treinamento e baixa taxa de acerto no conjunto de teste). Notou-se que a introdução do *dropout*, da maneira como foi realizada, reduziu o desempenho de redes com poucas camadas.

A fim de avaliar a rede rasa obtida, foi criado um *baseline* para referência. Esse *baseline* simplesmente consistiu em modelo SVM treinado com vetor de características extraído do conjunto de dados através do modelo ResNet-50. A comparação resultou em uma acurácia 28,7% maior para a rede rasa, indicando que sua configuração e treinamento foram satisfatórios.

Um ponto evidenciado é a menor acurácia obtida não só pela rede rasa, mas também pelo *baseline*, em classificar os dois estágios mais leves da doença. No entanto, testes mostraram que mesmo reduzindo o problema para uma classificação binária, houve dificuldade em separar essas classes adjacentes (*Non Demented* x *Very Mild Demented* e *Very Mild Demented* x *Mild Demented*) e isso também é corroborado pela dificuldade da identificação do Alzheimer por médicos durante seus estágios iniciais.

Um próximo passo desse trabalho seria aplicar uma técnica de autosupervisão (ou no inglês, *self supervision*) para treinamento da rede dado que o conjunto de dados é limitado, característica comum entre os dados do domínio médico. Para isso, seria necessário treinar a rede para um outro propósito, fazendo com que o peso da rede sejam inicializados, para depois, fazer seu o refinamento (*fine tuning*). Sob essa mesma linha de raciocínio, poderia se utilizar um *autoencoder*.

Outra ideia é utilizar o fato das imagens consistem de diferentes fatias de ressonância magnética do cérebro, o que impõem um maior nível de complexidade ao problema, e agrupá-los em conjunto de dados nas diferentes fatias para o treinamento mais eficiente da rede.

Concluindo, visto que um dos maiores aliados no tratamento da doença de Alzheimer é sua detecção precoce, uma vez que ela é incurável até o presente momento, uma rede capaz de detectar a doença através de uma imagem de ressonância magnética com mais de 78% de acerto é uma grande ferramenta para o combate desse mal que aflige milhares de pessoas ao redor do mundo.

REFERÊNCIAS

- [1] The Nobel Prize in Physiology or Medicine 2003. **NobelPrize.org**, 2003. Disponível em: <https://www.nobelprize.org/prizes/medicine/2003/summary/>. Acesso em: 22 set. 2021.
- [2] BERNSTEIN, M. A; KING, K. F.; ZHOU, X. J. **Handbook of MRI Pulse Sequences**. Elsevier, 2004.
- [3] DAWSON, M. J. **Paul Lauterbur and the Invention of MRI**. 1. ed. MIT Press, 2013.
- [4] SUTHERLAND, I. E.; AIKEN, H. Raymond V. Damadian. **National Inventors Hall of Fame**. Disponível em: <https://www.invent.org/inductees/raymond-v-damadian>. Acesso em: 22 set. 2021.
- [5] GRAND VIEW RESEARCH. **Magnetic Resonance Imaging Market Size, Share & Trends Analysis Report By Architecture, By Field Strength, By Application (Brain & Neurological, Spine & Musculoskeletal), By End-use, By Region, And Segment Forecasts, 2021 - 2028**. San Francisco: Grand View Research, 2020.
- [6] FERNANDEZ, E. Medical Imaging Rates Continue to Rise Despite Push to Reduce Their Use. **UCSF**, 2019. Disponível em: <https://www.ucsf.edu/news/2019/09/415286/medical-imaging-rates-continue-rise-despite-push-reduce-their-use>. Acesso em: 22 set. 2021.
- [7] Number of examinations with magnetic resonance imaging (MRI) in selected countries as of 2019. **statista**, 2021. Disponível em: <https://www.statista.com/statistics/271470/mri-scanner-number-of-examinations-in-selected-countries/>. Acesso em: 22 set. 2021.
- [8] FRISONI, G. B. *et al.* The clinical use of structural MRI in Alzheimer disease. **Nat Rev Neurol**, v. 6, p. 67-77, 2010.
- [9] HIPPUS, H., NEUNDÖRFER, G. The discovery of Alzheimer's disease. **Dialogues Clin Neurosci**, p. 101-108, 2003.
- [10] SMITH, M. A. Alzheimer Disease. **International Review of Neurobiology**, v. 42, p. 1 - 54, 1988
- [11] REITZ, C., BRAYN, C., MAYEUX, R. Epidemiology of Alzheimer disease. **Nat Rev Neurol**, v. 7, p. 137–152, 2011.
- [12] Stages of Alzheimer's. **Alzheimer's Association**. Disponível em: <https://www.alz.org/alzheimers-dementia/stages>. Acesso em: 1 out. 2021.
- [13] What Happens to the Brain in Alzheimer's Disease?. **National Institute on Aging**, 2017. Disponível em: <https://www.nia.nih.gov/health/what-happens-brain-alzheimers-disease>. Acesso em: 1 out. 2021
- [14] LOCK, M. **The Alzheimer Conundrum**. Princeton University Press, 2013.
- [15] CASTELLANI, R. J., ROLSTON, R. K. Alzheimer disease. **Disease-a-month**, v. 56,9 , p. 484 - 546, 2010.
- [16] SINTOMAS E DOENÇAS Alzheimer: principais informações. **Vida Saudável O blog do Einstein**. 2021. Disponível em: <https://vidasaudavel.einstein.br/alzheimer-principais-informacoes/>. Acesso em: 1 out. 2021.
- [17] LONG, X., CHEN, K., JIANG, C., ZHANG, L. Prediction and classification of Alzheimer disease based on quantification of MRI deformation. **PLoS ONE**, 2017.
- [18] GRAUPE, D. **Principles of Artificial Neural Networks**. 3. ed. Singapura: World Scientific Publishing Company, 2013.
- [19] MCCULLLOCH, W. S.; PITTS W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, v. 5, p. 115 - 133, dez. 1943.

- [20] HEBB, D. O. **The Organization of Behavior: A Neuropsychological Theory**. 1. ed. Wiley, 1949.
- [21] ROSENBLATT, F. **The perceptron: A perceiving and recognizing automaton**. 1. ed. Ithaca: Cornell Aeronautical Laboratory. p. 460 - 461, 1957.
- [22] RUMELHART, D. E.; MCCLELLAND, J. L. **Parallel Distributed Processing - Explorations in the Microstructure of Cognition**. 1. ed. MIT Press, 1986.
- [23] PATEL, J. L.; RAMESH, K. G. Applications of artificial neural networks in medical science. **Current Clinical Pharmacology**, v. 2, p. 217 - 226, 2007.
- [24] SHIN, H. *et al.* Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. **IEEE Transactions on Medical Imaging**, p. 1285 - 1298, 2016.
- [25] LEIFER, B. P. Early Diagnosis of Alzheimer's Disease: Clinical and Economic Benefits. **Journal of the American Geriatrics Society**, v. 51, p. 281-288, 2003.
- [26] HALEY, W. E. The family caregiver's role in Alzheimer's disease. **Neurology**, p. 25 – 29, 1997
- [27] HARVEY, P. D. *et al.* The validation of a caregiver assessment of dementia: the Dementia Severity Scale. **Alzheimer disease and associated disorders**, p. 186 – 194, 2005.
- [28] ZHANG, Y. *et al.* Detection of subjects and brain regions related to Alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning. **Frontiers in Computational Neuroscience**, v. 9, p. 66, 2015.
- [29] Dataset_Alzheimer. **Kaggle**, 2021. Disponível em: <https://www.kaggle.com/yasserhessein/dataset-alzheimer>. Acesso em: 22 set. 2021.
- [30] PONTI, M. A. *et al.* Everything you wanted to know about Deep Learning for Computer Vision but were afraid to ask. **2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials**. p. 17 - 41, 2017.
- [31] What is Alzheimer's Disease. Alzheimer's Association. Disponível em: <https://www.alz.org/alzheimers-dementia/what-is-alzheimers>. Acesso em: 30 nov. 2021.
- [32] CASTRILLO, J. I.; OLIVER, S. G. **Systems Biology of Alzheimer's Disease**: alzheimer's as a systems-level disease involving the interplay of multiple cellular networks. Nova York: Springer Science+Business, 2016. 46 p. (Methods in Molecular Biology).
- [33] LEVY, S. New brain imaging agent is marker for progression of Alzheimer's. **MedCityNews**. 2016. Disponível em: <https://medcitynews.com/2016/05/brain-imaging-agent-alzheimer/>. Acesso em: 30 nov. 2021.
- [34] O'Brien, R. J.; Wong P. C. Amyloid precursor protein processing and Alzheimer's disease. **Annu Rev Neurosci**, p. 185-204, 2011.
- [35] CHEN, G. *et al.* Amyloid beta: structure, biology and structure-based therapeutic development. **Acta Pharmacologica Sinica**, p. 1205-1235, 2017.
- [36] APP gene. **MedlinePlus**, p. 1-5, 2020. Disponível em: <https://medlineplus.gov/download/genetics/gene/app.pdf>. Acesso em: 30 nov. 2021
- [37] HOENIG, M. C. *et al.* Networks of tau distribution in Alzheimer's disease. **BRAIN**, p. 568-581, 2018.
- [38] BINDER, L. I. *et al.* Tau, tangles, and Alzheimer's disease. **Biochimica et Biophysica Acta**, p. 216-223, 2005.
- [39] LIMA, F. R.; DECKNIS, G. T. Doença de Alzheimer: As perspectivas de se encontrar biomarcadores. **Neurociências em debate**. 2016. Disponível em: <http://cienciascognitiva.org/neuroemdebate/arquivos/3249>. Acesso em: 30 nov. 2021.

- [40] FALCO, A. *et al.* Doença de Alzheimer: Hipóteses etiológicas e perspectivas de tratamento. **Química Nova**, p. 63-80, v. 39, 2016. Disponível em: <https://www.scielo.br/j/qn/a/6QpByS45Z7qYdBtD5MTNcP>. Acesso em: 30 nov. 2021.
- [41] Earlier Diagnosis. **Alzheimer's Association**. Disponível em: https://www.alz.org/alzheimers-dementia/research_progress/earlier-diagnosis. Acesso em: 30 nov. 2021.
- [42] BARROS, E. P. S. **A Utilização de Ressonância Magnética no auxílio do Diagnóstico da Doença de Alzheimer**. Curso de Curso de Especialização em Imagenologia Biomédica, Instituto Nacional de Ensino Sociedade e Pesquisa, Recife, 2017.
- [43] MAZZOLA, A. A. Ressonância magnética: princípios de formação da imagem e aplicações em imagem funcional. **Revista Brasileira de Física Médica**, p. 117-129, 2009
- [44] NOWOGRODZKI, A. **The world's strongest MRI machines are pushing human imaging to new limits**. Nature, p. 24-28, 2018.
- [45] CHERRY, K. An Overview of the Different Parts of a Neuron. **verywell mind**. 2020. Disponível em: <https://www.verywellmind.com/structure-of-a-neuron-2794896>. Acesso em: 30 nov. 2021.
- [46] KROGH, A. What are artificial neural networks? **Nature Biotechnology**, p. 195-197, 2008.
- [47] SZYMIK, B. Neuron Anatomy. **Ask a Biologist**, 2011. Disponível em: <https://askabiologist.asu.edu/neuron-anatomy>. Acesso em: 30 nov. 2021.
- [48] SHARMA, S.; SHARMA, S. Activation Function in Neural Networks. **International Journal of Engineering Applied Sciences and Technology**, p. 310-316, v.4, 2020.
- [49] IBM Cloud Education. Neural Networks. **IBM Cloud Learn Hub**, 2020. Disponível em: <https://www.ibm.com/cloud/learn/neural-networks>. Acesso em: 30 nov. 2021.
- [50] WALCZAK, S.; CERPA, N. **Encyclopedia of Physical Science and Technology: Artificial Neural Networks**. Boulder: Academic Press, 2003, p. 631-645.
- [51] Multi-Class Neural Networks: Softmax. Machine Learning Crash Course. Disponível em: <https://developers.google.com/machine-learning/crash-course/multi-class-neural-networks/softmax>. Acesso em: 30 nov. 2021
- [52] KOSTADINOV, S. Understanding Backpropagation Algorithm. **Towards data science**. 2019. Disponível em: <https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd>. Acesso em: 30 nov. 2021
- [53] NICHOLS, J. A.; CHAN, H. W. H.; BAKER, M. A. B. Machine learning: applications of artificial intelligence to imaging and diagnosis. **Biophysical reviews**, p.111-118, 2019.
- [54] BROWNLEE, J. Overfitting and Underfitting with Machine Learning Algorithms. **Machine Learning Mastery**. 2019. Disponível em: <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>. Acesso em: 30 nov. 2021.
- [55] KASIRAJAN, A. UNDERFIT and OVERFIT Explained. **Medium**. 2020. Disponível em: <https://medium.com/@minions.k/underfit-and-overfit-explained-8161559b37db>. Acesso em: 30 nov. 2021.
- [56] SRIVASTAVA, N. *et al.* Dropout: A Simple Way to Prevent Neural Networks from Overfitting. **Journal of Machine Learning Research 15**, p. 1929-1958, 2014.
- [57] Capítulo 23 – Como Funciona o Dropout? **Deep Learning Book**. Disponível em: <https://www.deeplearningbook.com.br/capitulo-23-como-funciona-o-dropout/>. Acesso em: 11 jul. 2022.
- [58] GRANDINI, M.; BAGLI, E.; VISANI, G. Metrics for Multi-Class Classification: An Overview. **ArXiv**, 2020.

- [59] SHMUELI, B. Multi-Class Metrics Made Simple, Part I: Precision and Recall. *Towards Data Science*. 2019. Disponível em: <https://towardsdatascience.com/multi-class-metrics-made-simple-part-i-precision-and-recall-9250280bddc2>. Acesso em: 12 jul. 2022.
- [60] RUSK, N. Deep Learning. *Nature Methods*, p. 35, v. 31, 2016.
- [61] NIELSEN, M. **Neural Networks and Deep Learning**. 2019. Disponível em: <http://neuralnetworksanddeeplearning.com/>. Acesso em: 14 dez. 2021.
- [62] BENGIO, Y.; GOODFELLOW, I.; COURVILLE, A. **Deep Learning**. MIT PRESS, 2016.
- [63] LECUN, Y.; BENGIO, Y.; HINTON, G. **Deep Learning**. *Nature*, p. 436-444, 2015.
- [64] YOU, Y. *et al.* Large Batch Optimization for Deep Learning: Training BERT in 76 minutes. **International Conference on Learning Representations**, 2020.
- [65] CHILIMBI, T. *et al.* Project Adam: Building an Efficient and Scalable Deep Learning Training System. **11th USENIX Symposium on Operating Systems Design and Implementation**, p. 571-582, 2014.
- [66] BOHRA, Y. The Challenge of Vanishing/Exploding Gradients in Deep Neural Networks. *Analytics Vidhya*. 2021. Disponível em: <https://www.analyticsvidhya.com/blog/2021/06/the-challenge-of-vanishing-exploding-gradients-in-deep-neural-networks/>. Acesso em: 14 dez. 2021.
- [67] WEISS, K.; KHOSHGOFTAAR, T. M.; WANG, D. A Survey of transfer learning. **Journal of Big Data**, 2016
- [68] DENG, J. *et al.* ImageNet: A Large-Scale Hierarchical Image Database. **CVPR09**, 2009.
- [69] BROWNLEE, J. Transfer Learning in Keras with Computer Vision Models. **Machine Learning Mastery**. 2019. Disponível em <https://machinelearningmastery.com/how-to-use-transfer-learning-when-developing-convolutional-neural-network-models/>. Acesso em: 15 dez. 2021.
- [70] TAKIMOGLU, A. Top Data Augmentation Techniques: Ultimate Guide for 2022. AI Multiple. 2022. Disponível em: <https://research.aimultiple.com/data-augmentation-techniques/>. Acesso em: 29 jun. 2022.
- [71] SHORTEN, C.; TAGHI, M. K. A survey on Image Data Augmentation for Deep Learning. **Journal of Big Data** 6, 2019.
- [72] GONDHALEKAR, A. Data Augmentation – Is it really necessary? *Analytics Vidhya*. 2020. Disponível em <https://medium.com/analytics-vidhya/data-augmentation-is-it-really-necessary-b3cb12ab3c3f>. Acesso em: 30 jun. 2022.
- [73] LI, S. Y. Automating Data Augmentation: Practice, Theory and New Direction. The Stanford AI Lab Blog, 2020. Disponível em <https://ai.stanford.edu/blog/data-augmentation/>. Acesso em: 30 jun. 2022.
- [74] ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. **International Conference on Engineering and Technology**, p. 1-6, 2017.
- [75] MISHRA, D. Convolution vs Correlation. **Towards data science**. 2019. Disponível em: <https://towardsdatascience.com/convolution-vs-correlation-af868b6b4fb5>. Acesso em: 30 nov. 2021.
- [76] BASAVARAJAIAH, M. 6 basic things to know about Convolution. Medium. 2019. Disponível em: <https://medium.com/@bdhuma/6-basic-things-to-know-about-convolution-dae5e1bc411>. Acesso em: 30 nov. 2021.
- [77] BASHA, S. H. S. *et al.* Impact of Fully Connected Layers on Performance of Convolutional Neural Networks for Image Classification. **Neurocomputing**, v. 378, p. 112-119, 2019.

- [78] Convolutional Neural Networks: An Introduction to Convolutional Neural Networks, 2018. Disponível em: <https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05>. Acesso em: 30 nov. 2021.
- [79] LECUN, Y. *et al.* Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, p. 2278-2324, 1998
- [80] KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. **Imagenet classification with deep convolutional neural networks**. Advances in Neural Information Processing System, p. 1106-1114, 2012.
- [81] SIMONYAN, K.; ZISSERMAN, A. Very Deep convolutional networks for large-scale image recognition. **CoRR**. 2014.
- [82] SZEGEDY, C. *et al.* Rethinking the inception architecture for computer vision. **CoRR**, 2015.
- [83] SZEGEDY, C. *et al.* Going deeper with convolutions. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. p. 1-9, 2015
- [84] HE, K. *et al.* Deep Residual Learning for Image Recognition. **CoRR**, 2015
- [85] DAS, S. CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more...Analytics Vidhya. 2017. Disponível em: <https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>. Acesso em: 30 nov. 2021.
- [86] HOWARD, A. G. *et al.* MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. **ArXiv**, 2017.
- [87] PUJARA, A. Image Classification With MobileNets. **Medium**. 2020. Disponível em: <https://medium.com/analytics-vidhya/image-classification-with-mobilenet-cc6fbb2cd470>. Acesso em: 04 ago. 2022.
- [88] SARKAR, A. Understanding Depthwise Separable Convolutions and the efficiency of MobileNets. **Towards Data Science**. 2021. Disponível em: <https://towardsdatascience.com/understanding-depthwise-separable-convolutions-and-the-efficiency-of-mobilenets-6de3d6b62503>. Acesso em: 04 ago. 2022.
- [89] RAJPURKAR, P. *et al.* CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, **ArXiv**, 2017.
- [90] HUANG, G. *et al.* Densely Connected Convolutional Networks. **ArXiv**, 2017.
- [91] RAJU, M. *et al.* Deep Learning Based Multilevel Classification of Alzheimer's Disease using MRI Scans. **IOP Conference Series: Materials Science and Engineering**. 2021.
- [92] SELVARAJU, R. R. *et al.* Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. IEEE International Conference on Computer Vision (ICCV), p. 618-626, 2017
- [93] HON, M.; KHAN, N. Towards Alzheimer's Disease Classification through Transfer Learning. **Bioinformatics and Biomedicine**, p. 1166-1169, 2017.
- [94] OASIS. Disponível em: <https://www.oasis-brains.org/>. Acesso em: 30 nov. 2021.
- [95] SARRAF, S. TOFIGHI, G. Classification of Alzheimer's Disease Structural MRI Data by Deep Learning Convolutional Neural Networks. **Cornell University**. 2016.
- [96] ADNI. Disponível em <http://adni.loni.usc.edu/>. Acesso em: 30 nov. 2021.
- [97] AL-KHUZAI, F. E. K.; BAYAT, O.; DURU, A. D. Diagnosis of Alzheimer Disease Using 2D MRI Slices by Convolutional Neural Network. **Applied Bionics and Biomechanics**, v. 2021, 2021
- [98] SARATXAGA, C. L. *et al.* MRI Deep Learning-Based Solution for Alzheimer's Disease Prediction. **Journal of Personalized Medicine**. 2021
- [99] BROWNLEE, J. A Gentle Introduction to Mini-Batch Gradient Descent and How to Configure Batch Size. Machine Learning Mastery. 2017. Disponível em

<https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descent-configure-batch-size/>. Acesso em: 30 nov. 2021.

- [100] BAE, J. B. *et al.* Identification of Alzheimer's disease using a convolutional neural network model based on T1-weighted magnetic resonance imaging. **Scientific reports** **10**, 2020
- [101] BI, X. *et al.* Artificial Intelligence-basedMRI Images for Brain in Prediction of Alzheimer's Disease. **Journal of Healthcare Engineering**. 2021
- [102] HELALY, H. A.; BADAWEY, M.; HAIKAL, A. Y. Deep Learning Approach for Early Detection of Alzheimer's Disease. **Cognitive Computation**. 2021
- [103] ODUSAMI, M. *et al.* Analysis of Features of Alzheimer's Disease: Detection of Early Stage from Functional Brain Changes in Magnetic Resonance Images Using a Finetuned ResNet18 Network. **Diagnostics**, 2021.
- [104] CHEN, B. Early Stopping in Practice: an example with Keras and TensorFlow 2.0. **Towards data science**. 2020. Disponível em: <https://towardsdatascience.com/a-practical-introduction-to-early-stopping-in-machine-learning-550ac88bc8fd>. Acesso em: 30 nov. 2021.
- [105] **Functional Neuroanatomy**. Disponível em: <https://www.neuroanatomy.ca/verticals.html>. Acesso em: 25 mar. 2022.