



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

# Segmentação de Lesões de Esclerose Múltipla aplicando Redes Neurais Convolucionais em Imagens de Ressonância Magnética

Clara Resende Maia

Monografia apresentada como requisito parcial  
para conclusão do Curso de Engenharia da Computação

Orientadora  
Prof.a Dr.a Roberta Barbosa Oliveira

Brasília  
2023



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

# Segmentação de Lesões de Esclerose Múltipla aplicando Redes Neurais Convolucionais em Imagens de Ressonância Magnética

Clara Resende Maia

Monografia apresentada como requisito parcial  
para conclusão do Curso de Engenharia da Computação

Prof.a Dr.a Roberta Barbosa Oliveira (Orientadora)  
CIC/UnB

Prof. Dr. Vinicius Ruela Pereira Borges    Prof. Dr. Pedro Garcia Freitas  
Universidade de Brasília                      Universidade de Brasília

Prof. Dr. João Luiz Azevedo de Carvalho  
Coordenador do Curso de Engenharia da Computação

Brasília, 15 de dezembro de 2023

# Dedicatória

Dedico este trabalho aos meus pais, que me apoiaram durante todos esses anos e me encorajaram a sempre aprender mais.

# Agradecimentos

Agradeço à minha família por todo o apoio e ajuda, especialmente à minha mãe, à minha irmã e ao meu pai, por serem pacientes e por me acompanharem nessa jornada.

Agradeço aos amigos de infância e aos amigos que conheci no curso pelo auxílio durante todos esses anos de estudos e pelos momentos de descontração.

Agradeço à minha professora orientadora Roberta Barbosa Oliveira, pela paciência, pelos ensinamentos e por me guiar durante toda a confecção desta monografia.

# Resumo

A esclerose múltipla é uma doença autoimune capaz de gerar lesões nos tecidos do sistema nervoso central e causar diversos impactos na coordenação motora de um indivíduo. Essa doença não tem cura ou causa bem definida. Estudos indicam que alguns fatores genéticos e ambientais aumentam o risco de incidência. O diagnóstico é auxiliado por exames de imagem de ressonância magnética e a detecção das lesões é realizada por profissionais da área médica. Esse processo pode ser demorado e propenso a erros. Com o intuito de apoiar o procedimento de diagnóstico, este trabalho propõe o uso de *Convolutional Neural Networks* (CNNs) para realizar a tarefa de segmentação de lesões de esclerose múltipla em imagens de ressonância magnética. São analisadas as arquiteturas UNet, ResUNet e Segnet, aliadas as técnicas de *transfer learning* e aumento de dados, a fim de verificar os modelos computacionais que obtêm o melhor desempenho. A base de dados utilizada inclui as sequências de ressonância magnética *Fluid-Attenuated Inversion Recovery* (FLAIR), T2, *Magnetization-Prepared Rapid Gradient-Echo* (MP-RAGE) e *Proton Density* (PD) e, para cada uma delas, foram realizados testes com cada modelo de CNN. Para avaliar os resultados foram empregadas as métricas coeficiente Dice, acurácia, sensibilidade e especificidade. Por meio da comparação e análise dos experimentos, foi possível concluir que a ResUNet, com aplicação do aumento de dados, obteve os melhores resultados para as imagens, alcançando um coeficiente Dice de 0.8268 para FLAIR, 0.7814 para T2, 0.7793 para MP-RAGE e 0.7867 para PD.

**Palavras-chave:** Aprendizado de Máquina, Redes Neurais Convolucionais, Segmentação de Imagens, Ressonância Magnética, Esclerose Múltipla

# Abstract

Multiple sclerosis is an autoimmune disease that generates damage to the central nervous system tissues and affects a person's motor coordination. This disease does not have a cure or a well defined cause. Studies show that some genetic and environmental factors increase the risk of contracting the disease. The diagnosis is supported by Magnetic Resonance Imaging (MRI) exams and the detection of lesions is made by medical professionals. This process can take a lot of time and errors may occur. In order to support the process of diagnosis, this study proposes the use of Convolutional Neural Networks (CNNs) to accomplish the task of multiple sclerosis lesion segmentation in MRI. An analysis of the architectures UNet, ResUNet and Segnet, combined with transfer learning and data augmentation techniques, is made in order to verify which computational models have the best performance. The dataset includes the MRI sequences Fluid-Attenuated Inversion Recovery (FLAIR), T2, Magnetization-Prepared Rapid Gradient-Echo (MP-RAGE) and Proton Density (PD) and, for each one of them, tests were made with each CNN model. The metrics employed to evaluate the results were Dice coefficient, accuracy, sensitivity and specificity. Through the analysis and comparison of the experiments, it was concluded that ResUNet, with the application of data augmentation, achieved the best results for the images and obtained a Dice coefficient of 0.8268 for FLAIR, 0.7814 for T2, 0.7793 for MP-RAGE and 0.7867 for PD.

**Keywords:** Machine Learning, Convolutional Neural Networks, Image Segmentation, Magnetic Resonance Imaging, Multiple Sclerosis

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Objetivos . . . . .	3
1.2	Estrutura do documento . . . . .	3
<b>2</b>	<b>Fundamentação Teórica</b>	<b>4</b>
2.1	Imagens de ressonância magnética . . . . .	4
2.2	Segmentação de imagens . . . . .	7
2.3	Aprendizado de máquina . . . . .	7
2.4	Rede Neural Convolutiva . . . . .	8
2.4.1	Camada Convolutiva . . . . .	9
2.4.2	Camada de Pooling . . . . .	10
2.4.3	Camada Totalmente Conectada . . . . .	11
2.4.4	Função de Ativação . . . . .	11
2.5	Arquiteturas de CNN . . . . .	12
2.5.1	UNet . . . . .	13
2.5.2	VGG16-UNet . . . . .	14
2.5.3	SegNet . . . . .	14
2.5.4	ResUNet . . . . .	15
2.6	Transfer Learning e Fine Tuning . . . . .	15
2.7	Aumento de dados . . . . .	15
2.8	Métricas de avaliação . . . . .	17
<b>3</b>	<b>Revisão de Literatura</b>	<b>19</b>
<b>4</b>	<b>Metodologia</b>	<b>24</b>
4.1	Base de dados . . . . .	24
4.2	Pré-processamento das imagens . . . . .	27
4.2.1	Redimensionamento . . . . .	27
4.2.2	Normalização de intensidades . . . . .	27
4.2.3	Aumento de dados . . . . .	28

4.3	Experimentos propostos . . . . .	28
4.3.1	UNet . . . . .	30
4.3.2	VGG16-UNet . . . . .	30
4.3.3	SegNet . . . . .	30
4.3.4	ResUNet . . . . .	31
4.4	Análise comparativa . . . . .	31
<b>5</b>	<b>Resultados</b>	<b>32</b>
5.1	Ferramentas utilizadas . . . . .	32
5.2	Experimento I: UNet . . . . .	32
5.3	Experimento II: VGG16-UNet . . . . .	37
5.4	Experimento III: SegNet . . . . .	41
5.5	Experimento IV: ResUNet . . . . .	45
5.6	Síntese dos resultados . . . . .	49
<b>6</b>	<b>Conclusão e trabalhos futuros</b>	<b>51</b>
	<b>Referências</b>	<b>53</b>



# Lista de Figuras

1.1	Comparação entre um neurônio saudável e um neurônio com deterioração da bainha de mielina. . . . .	1
2.1	Comportamento dos prótons durante um exame de ressonância magnética. . . . .	5
2.2	Um exemplo das principais sequências de imagens de ressonância magnética. . . . .	6
2.3	Exemplos das sequências de ressonância magnética MP-RAGE e PD, respectivamente [1]. . . . .	7
2.4	Exemplos de segmentação semântica de imagens. . . . .	8
2.5	Exemplo de estrutura de uma CNN. . . . .	9
2.6	Operação de convolução com uma entrada $4 \times 3$ e um filtro $2 \times 2$ , gerando como resultado uma saída $3 \times 2$ . . . . .	10
2.7	Operação de <i>max-pooling</i> com uma imagem $4 \times 4$ e um filtro $2 \times 2$ , gerando como resultado um mapa de características reduzido, de dimensões $2 \times 2$ . . . . .	11
2.8	Representação gráfica da função de ativação Sigmoid. . . . .	12
2.9	Representação gráfica da função de ativação ReLU. . . . .	12
2.10	Arquitetura da UNet. . . . .	13
2.11	Arquitetura da VGG16-UNet. . . . .	14
2.12	Esquema do modelo da ResUNet. . . . .	16
4.1	Diagrama da metodologia proposta. . . . .	25
4.2	Exemplos de imagens pré-processadas da base de dados para cada sequência de ressonância magnética [2]. . . . .	26
4.3	Exemplos de imagens geradas após a aplicação da técnica de aumento de dados. . . . .	29
4.4	Fluxograma dos experimentos realizados para cada tipo de imagem. . . . .	30
5.1	Curva de coeficiente Dice da UNet sem aumento de dados. . . . .	34
5.2	Curva de coeficiente Dice da UNet com aumento de dados. . . . .	35

5.3	Exemplos de segmentação de lesões de esclerose múltipla com a UNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	36
5.4	Exemplos de segmentação de lesões de esclerose múltipla com a UNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	37
5.5	Curva de coeficiente Dice da VGG16-UNet sem aumento de dados. . . . .	39
5.6	Curva de coeficiente Dice da VGG16-UNet com aumento de dados. . . . .	39
5.7	Exemplos de segmentação de lesões de esclerose múltipla com a VGG16-UNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	40
5.8	Exemplos de segmentação de lesões de esclerose múltipla com a VGG16-UNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	41
5.9	Curva de coeficiente Dice da SegNet sem aumento de dados. . . . .	43
5.10	Curva de coeficiente Dice da SegNet com aumento de dados. . . . .	43
5.11	Exemplos de segmentação de lesões de esclerose múltipla com a SegNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	44
5.12	Exemplos de segmentação de lesões de esclerose múltipla com a SegNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	45
5.13	Curva de coeficiente Dice da ResUNet sem aumento de dados. . . . .	47
5.14	Curva de coeficiente Dice da ResUNet com aumento de dados. . . . .	47
5.15	Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	48

5.16	Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	49
6.1	Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet e a técnica de aumento de dados para imagens FLAIR. É mostrada a sobreposição do contorno do <i>ground-truth</i> , em vermelho, e do contorno predito pelo modelo, em verde, na imagem original. . . . .	52

# Lista de Tabelas

3.1	Resumo dos trabalhos apresentados neste capítulo. . . . .	23
5.1	Resultados obtidos para a arquitetura UNet sem aumento de dados. . . . .	33
5.2	Resultados obtidos para a arquitetura UNet com aumento de dados. . . . .	33
5.3	Resultados obtidos para a arquitetura VGG16-UNet sem aumento de dados. . . . .	38
5.4	Resultados obtidos para a arquitetura VGG16-UNet com aumento de dados. . . . .	38
5.5	Resultados obtidos para a arquitetura SegNet sem aumento de dados. . . . .	42
5.6	Resultados obtidos para a arquitetura SegNet com aumento de dados. . . . .	42
5.7	Resultados obtidos para a arquitetura ResUNet sem aumento de dados. . . . .	46
5.8	Resultados obtidos para a arquitetura ResUNet com aumento de dados. . . . .	46
5.9	Resultado do coeficiente Dice após aplicar as arquiteturas UNet, VGG16-UNet, Segnet e ResUNet para cada sequência de ressonância magnética. . . . .	50

# Lista de Abreviaturas e Siglas

**ANFIS** *Adaptive Neuro-Fuzzy Inference System.*

**BET** *Brain Extraction Tool.*

**CNN** *Convolutional Neural Network.*

**DT** *Decision Tree.*

**DWT** *Discrete Wavelet Decomposition.*

**FLAIR** *Fluid-Attenuated Inversion Recovery.*

**HSL** *Hierarchical Specialization Learning.*

**ISBI** *International Symposium on Biomedical Imaging.*

**KNN** *K-Nearest Neighbors.*

**LCR** *Líquido Cefalorraquidiano.*

**LR** *Logistic Regression.*

**LST** *Lesion Segmentation Tool.*

**MICCAI** *Medical Image Computing and Computer Assisted Intervention.*

**MP-RAGE** *Magnetization-Prepared Rapid Gradient-Echo.*

**MP2RAGE** *Magnetization-Prepared 2 Rapid Acquisition Gradient Echo.*

**NIFTI** *Neuroimaging Informatics Technology Initiative.*

**PD** *Proton Density.*

**R-CNN** *Region-Based Convolutional Neural Network.*

**ReLU** *Rectified Linear Unit.*

**RNA** Rede Neural Artificial.

**SVM** *Support Vector Machine.*

**TE** Tempo de Eco.

**TPCNN** *Two-Path Convolutional Neural Network.*

**TR** Tempo de Repetição.

# Capítulo 1

## Introdução

O sistema nervoso central é formado por um conjunto de células e tecidos responsáveis pela transmissão de estímulos do ambiente interno e externo. Isso possibilita ao ser humano a execução de ações, como andar e falar. O sistema nervoso é constituído de células chamadas neurônios. Um neurônio é composto por três estruturas principais: o corpo celular, os dendritos e o axônio. O axônio é revestido por uma camada protetora, chamada bainha de mielina, retratada na Figura 1.1, pela célula saudável.

Existem diversas doenças que podem afetar o sistema nervoso central e causar limitações severas ao ser humano. Dentre elas destaca-se a esclerose múltipla, uma doença autoimune e inflamatória [3]. Nela, os neurônios são atacados pelo sistema imune do próprio organismo, causando a destruição da bainha de mielina e até mesmo do axônio, como pode ser visto na Figura 1.1, na célula afetada. Isso gera lesões nos tecidos do sistema nervoso central, afetando o cérebro, nervos e até mesmo a coluna espinhal. De acordo com a Organização Mundial da Saúde [4], a doença atinge 1.8 milhão de pessoas no mundo, sendo mais comum em jovens adultos, afetando principalmente a população feminina.

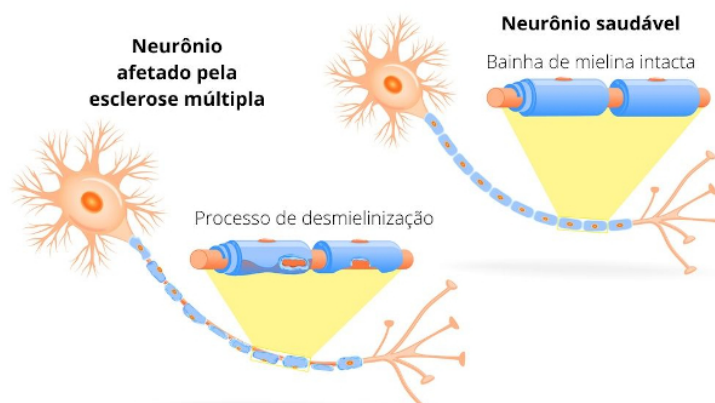


Figura 1.1: Comparação entre um neurônio saudável e um neurônio com deterioração da bainha de mielina (Fonte: [5]).

Pessoas com esclerose múltipla podem desenvolver sintomas variados, cuja severidade depende da localização e da extensão dos danos nos tecidos. Alguns deles incluem problemas de visão, dificuldade em andar, dormência e fraqueza dos membros e cansaço excessivo. A doença apresenta três estágios principais [6]:

- Recorrente-remitente: é a forma mais comum da doença e atinge cerca de 85% dos pacientes. Este estágio é caracterizado pela ocorrência de surtos intermitentes, seguidos por um período de remissão no qual os sintomas podem desaparecer;
- Primariamente progressivo: esse estágio começa a partir de sintomas iniciais e progride com o passar do tempo, gerando uma piora na condição do paciente. Não há períodos de remissão ou surtos bem definidos;
- Secundariamente progressivo: esse estágio inicia-se a partir da esclerose múltipla recorrente-remitente, seguido por uma progressão lenta da doença e diminuição da ocorrência de surtos.

A esclerose múltipla não possui uma causa definida, mas existem fatores genéticos e ambientais que aumentam o risco de incidência. Pessoas com histórico de esclerose múltipla na família têm maior suscetibilidade de contrair a doença. Baixos níveis de vitamina D também são tidos com uma possível causa. Devido a isso, regiões próximas ao equador têm menor incidência da doença. Outros fatores incluem o tabagismo e a contração prévia do vírus Epstein–Barr, causador da mononucleose [3].

A esclerose múltipla não tem cura, porém existem tratamentos eficazes que podem atenuar ou estabilizar os sintomas caso seja detectada em seu estágio inicial. Não existe um método de diagnóstico definitivo para a doença. No entanto, há testes que podem auxiliar no processo, como exames de imagem de ressonância magnética, capazes de mostrar lesões no cérebro e na coluna espinhal. Após a realização desses exames, um profissional da área médica identifica manualmente as regiões impactadas nas imagens. Esse processo pode ser demorado e desafiador para os médicos, devido ao número elevado de imagens que devem ser analisadas, e sujeito a erros [7] [8].

Estudos na área de inteligência artificial propõem técnicas e ferramentas capazes de automatizar esse processo de delimitação de lesões. La Rosa *et al.* [9], a fim de demarcar lesões de esclerose múltipla em imagens de ressonância, faz uso de uma *Convolutional Neural Network* (CNN). A CNN é um modelo computacional que simula o funcionamento do cérebro humano, sendo capaz de extrair múltiplas características de imagens [10]. Esse modelo pode ser aplicado na tarefa de segmentação das lesões, processo no qual as imagens são divididas em diversas partes. Em seguida, é realizado o reconhecimento de cada uma delas, com o intuito de averiguar se elas são parte de uma lesão ou não [11]. Além disso,



os autores também exploram a técnica de aprendizado de máquina de aumento de dados, que ajuda a melhorar a eficiência de uma rede neural [12].

Utilizando esses algoritmos, é possível auxiliar no diagnóstico da doença de forma rápida ao fornecer ao especialista uma segunda opinião. Estudos mostram a eficácia da CNN aplicada na área médica, como pode ser visto nas obras de Cheng e Lam [13] e Hashemi *et al.* [14]. Em seu trabalho, La Rosa *et al.* [9] aplicam o modelo UNet, aliado ao aumento de dados, em uma base formada por diferentes imagens de ressonância magnética. Neste trabalho, são empregados os modelos UNet, VGG16-UNet, SegNet e ResUNet, em conjunto com o *transfer learning* e o aumento de dados, para realizar a segmentação de lesões de esclerose múltipla em imagens de ressonância magnética.

## 1.1 Objetivos

O objetivo deste trabalho é realizar a segmentação de lesões de esclerose múltipla em imagens de ressonância magnética a partir da análise de técnicas de aprendizado de máquina e diferentes modelos de CNN, a fim de auxiliar no diagnóstico da doença. Para esse propósito, os seguintes objetivos específicos devem ser considerados:

- Analisar a eficácia das técnicas de *transfer learning* e aumento de dados no desempenho dos modelos de CNN UNet, VGG16-UNet, SegNet e ResUNet;
- Verificar quais arquiteturas de CNN obtêm melhores resultados na resolução do problema;
- Avaliar o desempenho dos modelos de CNN aplicados em diferentes sequências de ressonância magnética.

## 1.2 Estrutura do documento

O Capítulo 2 aborda os principais conceitos necessários para a compreensão deste trabalho. O Capítulo 3 detalha estudos realizados sobre o problema, citando técnicas aplicadas e resultados obtidos. O Capítulo 4 explica a metodologia aplicada nos procedimentos experimentais. O Capítulo 5 descreve os resultados obtidos e uma análise das técnicas empregadas. Por fim, o Capítulo 6 apresenta as conclusões sobre os métodos aplicados e trabalhos futuros.

# Capítulo 2

## Fundamentação Teórica

O presente capítulo apresenta os principais conceitos teóricos necessários para compreender este trabalho e os métodos utilizados. Na Seção 2.1 são abordados detalhes sobre a aquisição e o uso de imagens de ressonância magnética. As Seções 2.3 e 2.4 explicam o que é Aprendizado de Máquina e Redes Neurais Convolucionais. Detalhes sobre as técnicas aplicadas no Capítulo 4 são encontrados nas Seções 2.5, 2.6 e 2.7. Por fim, a Seção 2.8 descreve as métricas de avaliação utilizadas nos experimentos.

### 2.1 Imagens de ressonância magnética

A ressonância magnética é um exame de imagem não invasivo capaz de gerar imagens detalhadas de partes do corpo humano, incluindo órgãos, tecidos e vasos sanguíneos. Esse exame é frequentemente utilizado para detectar ou monitorar doenças e auxiliar em diagnósticos. Uma das vantagens desse procedimento é que, diferentemente da radiografia, a captura das imagens não utiliza radiação ionizante, que em grandes quantidades pode causar alguns efeitos colaterais, como náuseas. Outra vantagem dos exames de ressonância magnética é a possibilidade de visualizar a anatomia do cérebro em três planos distintos: axial, sagital e coronal<sup>1</sup> [15].

A captura das imagens é realizada por meio de um aparelho que emite um campo magnético e faz com que os prótons do corpo do paciente se alinhem a ele. Em seguida, ondas de radiofrequência são emitidas pelo equipamento a fim de desalinhar os prótons, e são desligadas após um tempo. Quando se realinham ao campo magnético, os prótons emitem ondas de radiofrequência responsáveis por formar as imagens [16], fenômeno representado pela Figura 2.1. A intensidade das ondas de retorno varia de acordo com os tecidos do corpo.

---

<sup>1</sup>O plano anatômico axial divide o cérebro nas partes superior e inferior; o sagital divide o cérebro nas metades direita e esquerda; o plano coronal divide o cérebro nas partes anterior e posterior.

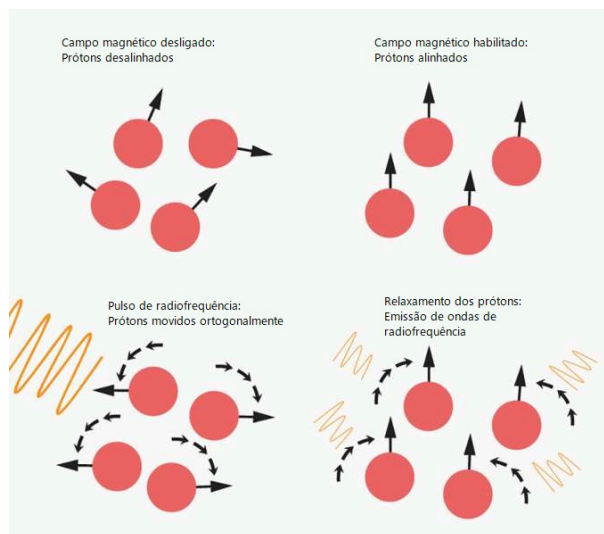


Figura 2.1: Comportamento dos prótons durante um exame de ressonância magnética (Fonte: [17]).

A partir da variação das ondas emitidas e coletadas pelo aparelho, é possível formar imagens com uma aparência específica. Os pulsos de radiofrequência podem ser descritos por duas variáveis: o Tempo de Repetição (TR) corresponde ao intervalo de tempo entre sequências de ondas emitidas sucessivamente em um mesmo local e o Tempo de Eco (TE) é o tempo entre a emissão do pulso e a recepção do sinal de eco. Ademais, existe outro conceito relevante no estudo das imagens de ressonância magnética: o tempo de relaxamento. Cada tecido tem um tempo de relaxamento diferente, sendo possível calcular esta métrica de duas formas: o tempo necessário para que um próton desalinhado se realinhe ao campo magnético é chamado relaxamento T1, enquanto que o tempo necessário para um próton excitado atingir o equilíbrio é chamado de T2.

A variação na quantidade de pulsos e gradientes de radiofrequência, aplicados na aquisição das imagens, é uma configuração conhecida como sequência de ressonância magnética. As sequências mais comuns são T1, T2, *Fluid-Attenuated Inversion Recovery* (FLAIR) e *Proton Density* (PD), e elas podem ser descritas pelas variáveis mencionadas anteriormente (TR e TE). A sequência T1 possui TR e TE curtos e suas características são predominantemente determinadas pelo relaxamento T1 dos tecidos. Nessas imagens, o Líquido Cefalorraquidiano (LCR) tem uma aparência mais escura e tecidos adiposos são apresentados com mais brilho, aspectos visíveis na Figura 2.2. Por outro lado, a sequência T2 possui TR e TE longos e as características deste tipo de imagem são definidas pela propriedade T2 dos tecidos. O LCR é exibido com destaque nessas imagens, assim como os tecidos adiposos [15]. A segunda imagem da Figura 2.2 (T2-weighted) exhibe esses detalhes.

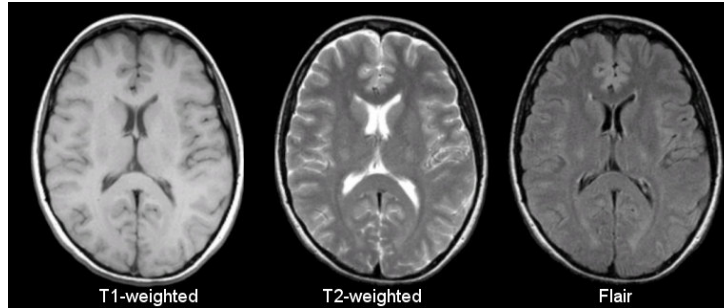


Figura 2.2: Um exemplo das principais sequências de imagens de ressonância magnética (Fonte: [15]).

A sequência FLAIR é semelhante à T2, porém o TR e o TE são bem mais longos. Como consequência, o LCR aparece nas imagens com um aspecto mais escuro, como pode ser visto na Figura 2.2, mas lesões e anormalidades são realçadas. Em virtude disso, imagens FLAIR têm mais impacto na detecção de lesões de esclerose múltipla [18]. Sequências do tipo PD possuem TR longo e TE curto, minimizando ambas as propriedades T1 e T2. Nessas imagens, o LCR é exibido com um aspecto acinzentado, como é visto na Figura 2.3.

Outra sequência de ressonância magnética relevante no estudo de doenças cerebrais é a *Magnetization-Prepared Rapid Gradient-Echo* (MP-RAGE). Ela é muito popular pois captura os tecidos com alto contraste, providenciando uma visão clara da anatomia do cérebro [19]. A Figura 2.3 mostra um exemplo dessa imagem. Nessa sequência, a propriedade T1 é predominante e, assim, suas imagens são frequentemente utilizadas na classificação de tecidos. Nos experimentos deste trabalho, foram utilizadas as sequências FLAIR, T2, MP-RAGE e PD.

Imagens de ressonância magnética geralmente são armazenadas em arquivos no formato *Neuroimaging Informatics Technology Initiative* (NIfTI). Esse formato foi adotado como padrão pela área de neuroimagem para armazenar imagens do cérebro. Arquivos NIfTI são capazes de armazenar os dados da ressonância e o seu cabeçalho em um único arquivo com extensão *.nii* [20]. Um único arquivo pode conter diversos *slices* e eles podem ser convertidos em imagens individuais do cérebro por meio das bibliotecas NiBabel<sup>2</sup> e NiLearn<sup>3</sup>.

---

<sup>2</sup><https://nipy.org/nibabel/>

<sup>3</sup><https://nilearn.github.io/stable/index.html>

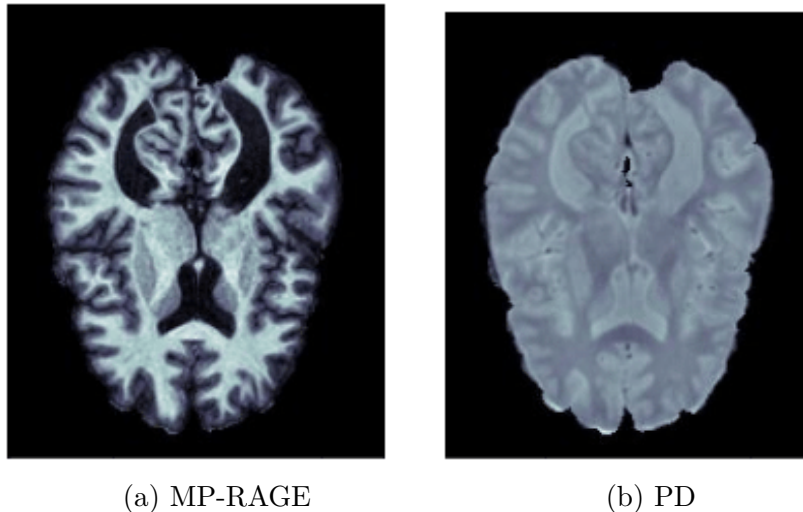


Figura 2.3: Exemplos das sequências de ressonância magnética MP-RAGE e PD, respectivamente [1].

## 2.2 Segmentação de imagens

Um dos procedimentos mais comuns na área de visão computacional é a segmentação de imagens. Esse processo consiste em dividir uma imagem em diversas partes e atribuir um rótulo para cada uma delas, a fim de identificar o que elas representam no contexto geral [11]. A segmentação tem muito destaque na análise de imagens médicas pois permite detectar doenças e extrair tecidos.

Para realizar uma segmentação, cada *pixel* é classificado com um rótulo único. Isso é feito para todos os *pixels* da imagem e é conhecido como segmentação semântica [11]. Por isso, esse processo é mais desafiador e dispendioso em comparação à classificação, que atribui um único rótulo a toda a imagem. A Figura 2.4 exibe alguns exemplos da segmentação semântica de imagens. Este trabalho propõe o uso da segmentação semântica para extrair lesões de esclerose múltipla de imagens de ressonância magnética, a fim de verificar a localização e dimensões das lesões e destacá-las de forma generalizada, sem diferenciá-las uma da outra.

## 2.3 Aprendizado de máquina

O aprendizado de máquina é uma subárea da Inteligência Artificial que propõe o uso de algoritmos capazes de aprender por meio de dados. Eles são alimentados com uma base de dados e aplicados na solução de problemas complexos, como a classificação de imagens em determinadas categorias ou a detecção de anomalias [12]. Um algoritmo de aprendizado

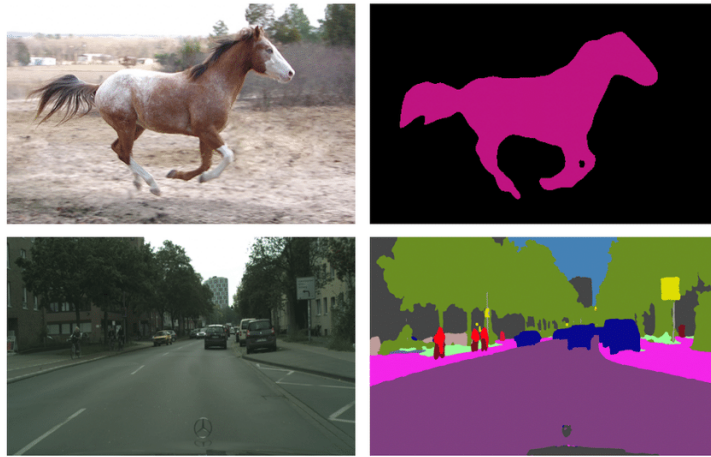


Figura 2.4: Exemplos de segmentação semântica de imagens (Fonte: [21]).

de máquina aprende por meio da experiência, que pode ser amplamente classificada como supervisionada ou não supervisionada [12].

Um algoritmo de aprendizado supervisionado é treinado com uma base de dados na qual cada exemplo é associado a rótulos predefinidos. O rótulo é o resultado esperado que deve ser obtido após o processamento. Considerando um exemplo  $x$  com um rótulo  $y$ , o algoritmo tenta prever o valor de  $y$  a partir do dado  $x$  por meio da observação de ambas as informações. Nesse tipo de aprendizado, os rótulos atuam como um instrutor ensinando ao algoritmo o que ele deve fazer [12].

Já no aprendizado não supervisionado, o treinamento é realizado por meio de uma base de dados sem nenhum rótulo. O algoritmo analisa o dado  $x$  e tenta extrair o máximo de características para prever  $y$ . O aprendizado não supervisionado tenta encontrar padrões sem nenhum instrutor para ensinar o que deve ser feito.

Alguns exemplos de aplicações de aprendizado supervisionado são técnicas de regressão e classificação; a técnica de agrupamento é característica do aprendizado não supervisionado. Neste trabalho, todos os experimentos utilizam o aprendizado supervisionado pois esta é a modalidade aplicada em CNNs.

## 2.4 Rede Neural Convolutacional

A área de aprendizado de máquina progrediu significativamente nos últimos anos e possibilitou o surgimento de novas técnicas de aprendizado com aplicações em diversas tarefas complexas, como o reconhecimento de padrões em imagens. Dentre essas técnicas, destaca-se a Rede Neural Artificial (RNA), um sistema computacional inspirado no funcionamento do sistema nervoso humano [10].

A RNA, assim como o cérebro humano, possui neurônios, ou nós, que são interconectados e capazes de realizar o processamento de dados. Cada nó possui o seu próprio peso e viés; quando ele recebe um valor de entrada, o processamento considera esse peso e viés na equação que calcula o valor de saída. Se a saída estiver acima do limite especificado o nó será ativado e enviará o resultado para a próxima camada [12]. Esse cálculo é realizado pela função de ativação, cujos detalhes podem ser encontrados na Subseção 2.4.4.

Existem diferentes tipos de redes neurais, como as CNNs. A CNN, é um tipo de RNA especializada no processamento de dados com topologia semelhante a uma matriz, como imagens. Uma CNN é composta por um número definido de camadas de processamento, capazes de extrair múltiplas características dos dados de entrada com diferentes níveis de abstração. Em 1989, LeCun *et al.* [10] desenvolveu uma CNN conhecida como LeNet-5, para a tarefa de reconhecimento de dígitos manuscritos. Apesar do sucesso desse modelo, CNNs ainda não tinham um bom desempenho em tarefas complexas devido à falta de bases de dados rotuladas e algoritmos inovadores. Em 2012, o desenvolvimento da arquitetura AlexNet, por Krizhevsky *et al.* [10], impulsionou a criação de diversos outros modelos com aplicações em áreas variadas.

As CNNs empregam uma operação de convolução em pelo menos uma de suas camadas, mas podem conter também operações de *pooling* e camadas totalmente conectadas, como a arquitetura da LeNet-5. A Figura 2.5 exemplifica um modelo conceitual e como essas camadas são dispostas. Nas Subseções 2.4.1, 2.4.2 e 2.4.3, esses três tipos de camadas serão explicados com mais detalhes.

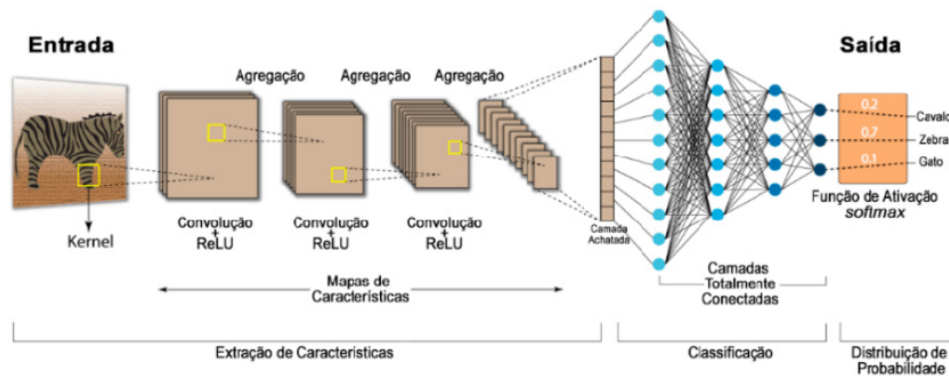


Figura 2.5: Exemplo de estrutura de uma CNN (Fonte: [22]).

## 2.4.1 Camada Convolutacional

A camada convolutacional é o elemento principal na estrutura de uma CNN. Ela é formada por filtros, ou *kernels*, e imagens de entrada e gera como resultado um mapa de características, após a operação de convolução [10].

O filtro consiste em uma matriz de dimensão  $n$ , na qual cada valor corresponde a um peso. Os pesos dos filtros de uma CNN são inicializados com valores aleatórios no início de cada treinamento. A convolução é realizada por meio da sobreposição do filtro na imagem; neste momento, os pesos do filtro são multiplicados pelos valores de cada *pixel* e somados no final. O resultado dessa operação compõe o mapa de características. Em seguida, o filtro é deslocado uma coluna para a direita e/ou uma linha para baixo, até que ele tenha percorrido toda a imagem.

A Figura 2.6 retrata o processo de convolução de uma imagem  $4 \times 3$  e um filtro  $2 \times 2$ . Os valores de entrada ( $a, b, \dots$ ) são multiplicados pelos pesos do filtro ( $w, x, \dots$ ), respectivamente, e depois somados para gerar cada *pixel* da imagem de saída ( $aw + bx, \dots$ ).

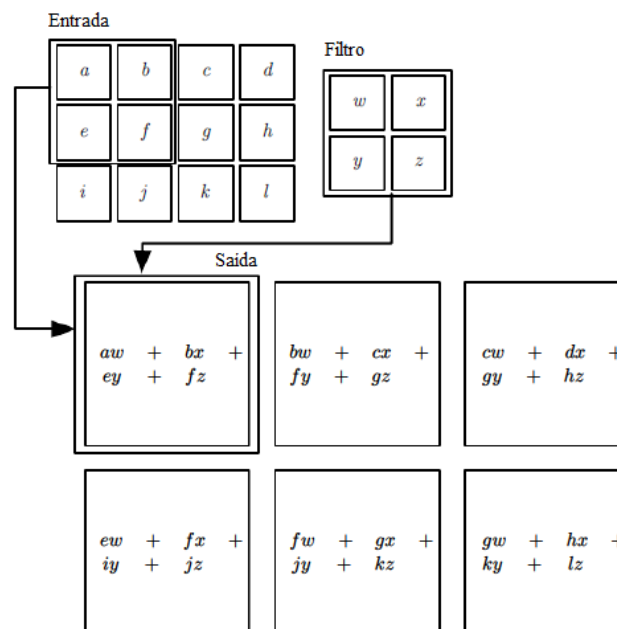


Figura 2.6: Operação de convolução com uma entrada  $4 \times 3$  e um filtro  $2 \times 2$ , gerando como resultado uma saída  $3 \times 2$  (Fonte: [12]).

## 2.4.2 Camada de Pooling

A camada de *Pooling* é frequentemente utilizada após a camada de convolução e tem como objetivo reduzir o mapa de características, de forma que sejam mantidas as informações mais relevantes. Essa camada aplica um filtro de dimensão  $n$  sobre a imagem de saída da camada anterior e realiza a operação de *pooling* até percorrer toda a matriz. Existem diferentes técnicas de *pooling* que podem ser implementadas em diferentes camadas. As mais comuns são o *max-pooling* e o *min-pooling* [10].



O *max-pooling*, mostrado na Figura 2.7, é a técnica mais popular em modelos de CNN e consiste em capturar o maior valor da região de sobreposição entre o filtro e a imagem. Um exemplo de como é realizada a operação de *max-pooling* é retratado na Figura 2.7.

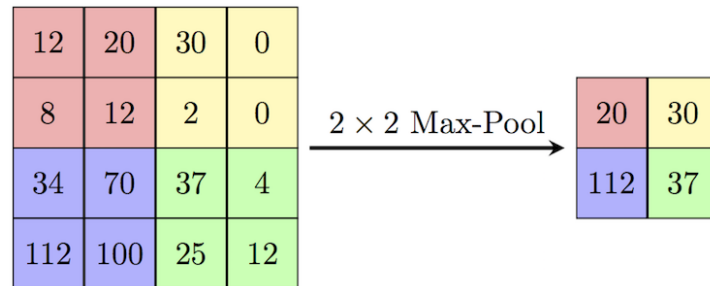


Figura 2.7: Operação de *max-pooling* com uma imagem 4x4 e um filtro 2x2, gerando como resultado um mapa de características reduzido, de dimensões 2x2 (Fonte: [23]).

### 2.4.3 Camada Totalmente Conectada

É comum que a última camada de uma CNN seja uma camada totalmente conectada, frequentemente utilizada em modelos de classificação. Ela recebe como valor de entrada a saída da última camada de convolução ou *pooling* e retorna a probabilidade de uma imagem pertencer a uma categoria.

### 2.4.4 Função de Ativação

A função de ativação é responsável por avaliar se um nó da rede neural será ativado ou não. Essa decisão depende do valor de entrada, calculado a partir da soma dos pesos e do viés do nó. Camadas de ativação geralmente são adicionadas à CNN após as camadas de convolução. O seu comportamento não linear permite que o modelo aprenda tarefas mais complexas [10]. As funções de ativação mais comuns são a Sigmoid e a *Rectified Linear Unit* (ReLU).

#### Sigmoid

A função de ativação Sigmoid normaliza os valores de entrada, representados por  $x$ , transformando-os em um número real entre 0 e 1, por meio da expressão matemática na Equação 2.1. A sua representação no gráfico possui o formato de uma letra S, como pode ser visto na Figura 2.8.

$$f(x)_{Sigmoide} = \frac{1}{1 + \exp^{-x}} \quad (2.1)$$

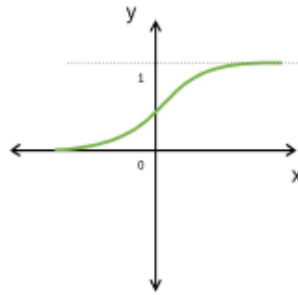


Figura 2.8: Representação gráfica da função de ativação Sigmoide (Fonte: [10]).

## ReLU

A função ReLU é a mais comum nas camadas de convolução de CNNs [10]. Ela converte todos os valores de entrada, representados por  $x$ , em números reais positivos e pode ser retratada pela Equação 2.2. Se o valor de entrada é um número menor que 0, a função retorna 0. Uma de suas vantagens é seu baixo custo computacional.

$$f(x)_{ReLU} = \max(0, x) \quad (2.2)$$

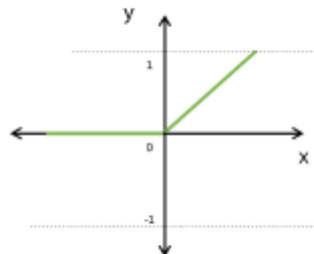


Figura 2.9: Representação gráfica da função de ativação ReLU (Fonte: [10]).

## 2.5 Arquiteturas de CNN

Nesta seção, são apresentados detalhes sobre as arquiteturas das CNNs utilizadas neste trabalho: UNet, VGG16-UNet, SegnNet e ResUNet. Cada modelo tem características específicas e emprega diferentes operações na sua estrutura.

## 2.5.1 UNet

CNNs são frequentemente aplicadas em problemas de classificação e fornecem como resultado a categoria à qual a imagem pertence, também conhecido como rótulo. Em 2015, Ronneberger *et al.* [24] desenvolveram um novo modelo de CNN chamado UNet para o processamento de imagens da área médica. A UNet foi criada para realizar a segmentação semântica de imagens, sendo capaz de localizar anormalidades por meio da classificação de cada *pixel* individualmente.

A UNet possui uma arquitetura em formato de U, como pode ser visto na Figura 2.10, e é considerada uma rede simétrica. Ela é composta por um caminho de contração, referente à parte esquerda da figura, e um caminho de expansão, que corresponde à parte direita. Cada caminho é formado por quatro blocos de camadas, conectados por uma ponte.

O caminho de contração possui a arquitetura típica de uma CNN. Cada conjunto de camadas é formado por duas convoluções, seguidas por uma função de ativação ReLU e uma operação de *Max-Pooling*, reduzindo as dimensões da imagem pela metade e abstraindo as suas informações. O caminho de expansão aumenta a resolução das imagens por meio de uma operação de convolução transposta, seguida por camadas de convolução e a concatenação com o mapa de características do caminho de contração. O resultado é uma máscara de segmentação com as mesmas dimensões que a imagem de entrada.

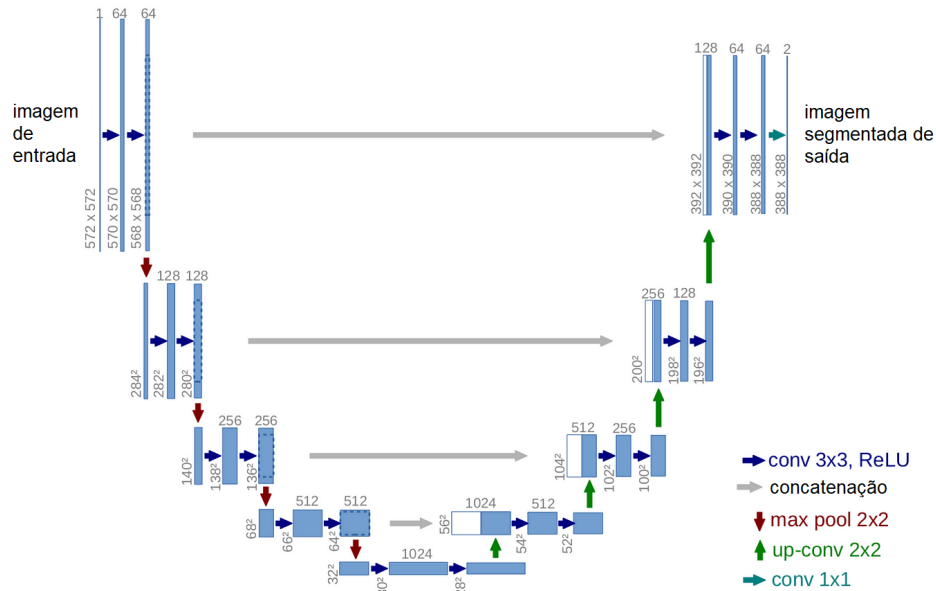


Figura 2.10: Arquitetura da UNet (Fonte: [24]).

## 2.5.2 VGG16-UNet

A VGG16-UNet, aplicada no trabalho de Cheng e Lam [13], possui uma arquitetura semelhante à UNet, porém ela conta com a aplicação da técnica de *transfer learning*, que auxilia no desempenho da rede. Esse modelo, também simétrico, utiliza no caminho de contração as camadas convolucionais da CNN VGG16, treinada na base de dados ImageNet<sup>4</sup>. Essas camadas já possuem os pesos predefinidos e, em virtude disso, são congeladas durante o treinamento para que os pesos não sejam alterados. O caminho de expansão é formado pelas camadas típicas da UNet adicionadas manualmente, que permanecem descongeladas e, por conseguinte, ajudam na adaptação do modelo à nova base de dados. Detalhes sobre essa implementação podem ser vistos na Figura 2.11.

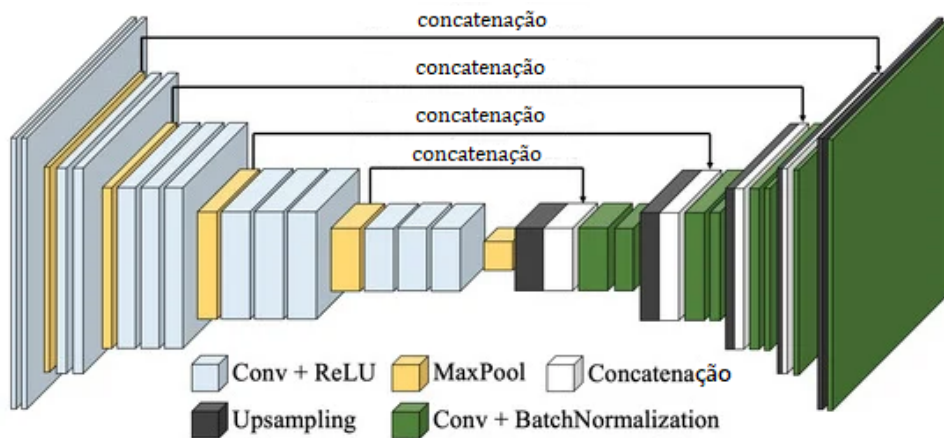


Figura 2.11: Arquitetura da VGG16-UNet (Fonte: [25]).

## 2.5.3 SegNet

Em 2017, Badrinarayanan *et al.* [26] desenvolveram uma nova arquitetura de CNN para segmentação semântica, chamada SegNet. Esse modelo também é simétrico, ou seja, é formado por um caminho de contração e de expansão. O caminho de contração tem uma arquitetura semelhante à rede neural VGG16 e, por isso, são utilizadas neste modelo as 13 camadas convolucionais da VGG16, pré-treinadas na base de dados ImageNet.

O diferencial da SegNet em relação a outros modelos, como a UNet, está no caminho de expansão da rede. Cada camada desta estrutura recebe os índices da operação de *max-pooling* da camada correspondente no caminho de contração. A partir desses índices, é realizada a operação de *upsampling*, responsável por aumentar a resolução da imagem de entrada. Essa abordagem reduz o número de parâmetros da rede e, conseqüentemente, o tempo de treinamento do modelo.

<sup>4</sup><https://www.image-net.org/>

### 2.5.4 ResUNet

Em 2018, Zhang *et al.* [27] propuseram um novo modelo de CNN a fim de realizar a extração de estradas em imagens aéreas. A nova arquitetura conhecida como *Deep Residual UNet*, ou apenas ResUNet, é baseada na arquitetura da UNet, mas contém unidades residuais na sua estrutura. Essa combinação permite que o modelo tenha facilidade no treinamento e seja capaz de propagar as informações pelas camadas sem muita degradação da imagem.

A ResUNet possui uma arquitetura simétrica, com um caminho de contração, um de expansão e uma ponte que conecta as duas partes. Todas as três estruturas são formadas por unidades residuais. Uma unidade é composta por duas camadas convolucionais, cada uma seguida de uma normalização em lote, ou *batch normalization*, e uma função de ativação ReLU. Por fim, uma operação de adição conecta a entrada e a saída da unidade. A Figura 2.12 exibe com mais detalhes a disposição das camadas.

## 2.6 Transfer Learning e Fine Tuning

Na área de aprendizado de máquina existem diversas técnicas que podem melhorar o desempenho de um algoritmo durante o aprendizado. Dentre essas técnicas encontra-se o *transfer learning*. Esse método consiste em aplicar um modelo previamente treinado em uma extensa base de dados na resolução de uma nova tarefa [12]. Assim, é possível reaproveitar os pesos já definidos e reduzir o tempo de treinamento. Modelos pré-treinados são capazes de boa generalização em tarefas, mesmo que a base de dados disponível seja pequena.

Quando o *transfer learning* é utilizado, as camadas do modelo pré-treinado podem ser congeladas para evitar que os pesos sejam alterados. O *fine tuning* descongela algumas camadas pré-treinadas para que seus pesos sejam atualizados durante o treinamento. Essa abordagem permite que o modelo se adapte aos novos dados e aprenda características específicas relacionadas à nova tarefa.

## 2.7 Aumento de dados

Para que uma CNN tenha um bom desempenho é necessário uma grande quantidade de dados. Bases de dados robustas geralmente não são fáceis de se obter, porém o modelo consegue generalizar melhor com mais artefatos, reduzindo o *overfitting* [12]. A técnica de aumento de dados, ou *data augmentation*, consegue minimizar esse problema por meio da criação de imagens artificiais.

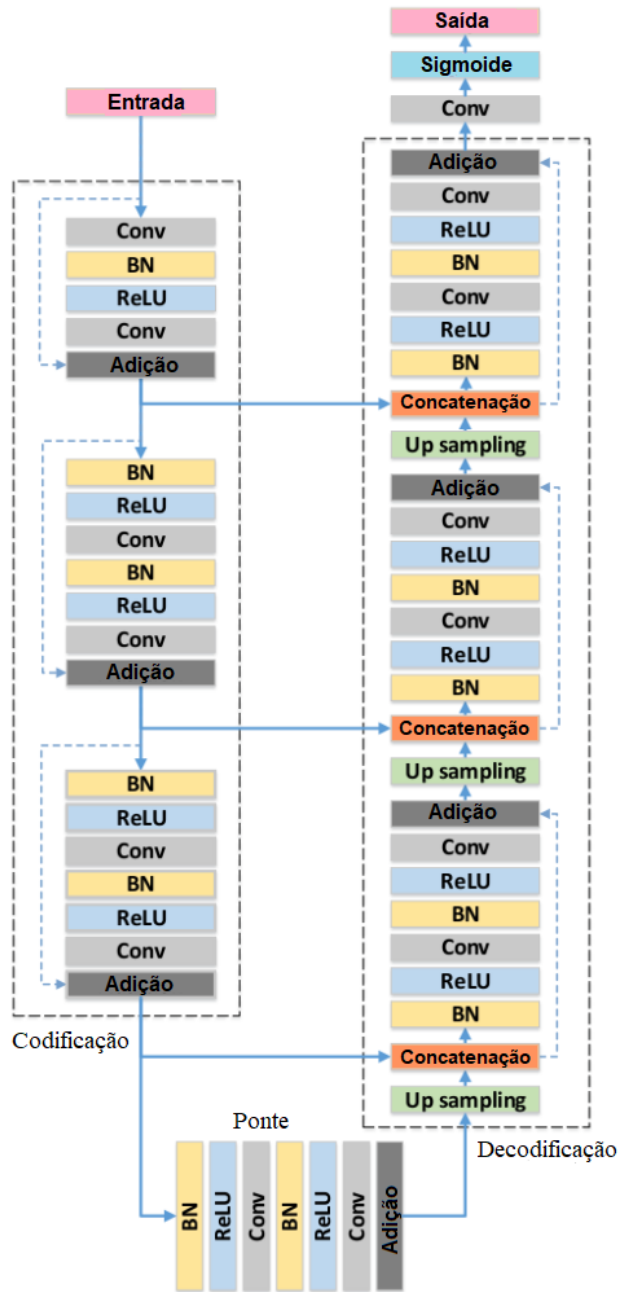


Figura 2.12: Esquema do modelo da ResUNet (Fonte: [27]).

Por meio do aumento de dados, é possível fazer pequenas alterações nas imagens disponíveis e adicioná-las ao conjunto de imagens existentes, aumentando significativamente o número de artefatos. As imagens originais permanecem na base de dados após a aplicação do método. Algumas das técnicas mais comuns são a rotação, o giro horizontal e vertical, *zoom*, recortes e mudanças no brilho e contraste [10].

## 2.8 Métricas de avaliação

Métricas de avaliação são necessárias para avaliar o desempenho de um algoritmo. Elas também permitem a comparação entre diferentes métodos de aprendizado de máquina e arquiteturas de CNN. Uma das formas mais simples e populares de visualizar o desempenho de uma rede neural é por meio da matriz de confusão. Ela engloba os diferentes resultados que a classificação dos *pixels* de uma imagem podem assumir ao serem comparados aos *pixels* do rótulo dessa mesma imagem. Uma matriz de confusão é formada por quatro quadrantes. Considerando a tarefa de segmentação de lesões de esclerose múltipla, esses quadrantes podem ser definidos da seguinte forma:

- Verdadeiro Positivo (VP): representa o número de *pixels* que foram corretamente classificados como parte de uma lesão;
- Falso Negativo (FN): representa o número de *pixels* que pertencem a uma lesão e foram erroneamente classificados como parte do plano de fundo ou outro tecido do cérebro que não corresponde a uma lesão;
- Falso Positivo (FP): representa o número de *pixels* que não pertencem a uma lesão e foram erroneamente classificados como tal;
- Verdadeiro Negativo (VN): representa o número de *pixels* que foram corretamente classificados como não pertencentes a uma lesão.

A partir dessas quatro variáveis é possível definir múltiplas métricas. As principais métricas utilizadas neste trabalho são definidas a seguir:

- **Acurácia:** corresponde ao total de *pixels* corretamente classificados, sejam eles parte de uma lesão ou não, dividido pelo número total de classificações:

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.3)$$

- **Sensibilidade:** corresponde ao número de *pixels* corretamente classificados como lesões dividido pelo total de *pixels* que deveriam ser classificados como lesões:

$$Sensibilidade = \frac{VP}{VP + FN} \quad (2.4)$$

- **Especificidade:** corresponde ao número de *pixels* corretamente classificados como parte do plano de fundo dividido pelo total de *pixels* que deveriam ser classificados como parte do plano de fundo:

$$Especificidade = \frac{VN}{VN + FP} \quad (2.5)$$

- **Coefficiente Dice:** é uma das métricas mais relevantes na segmentação de imagens médicas. Ela avalia a intersecção entre a imagem prevista (máscara de segmentação) e o rótulo da imagem. É calculada por meio da média harmônica das métricas Precisão e Revocação:

$$CoeficienteDice = \frac{2VP}{2VP + FP + FN} \quad (2.6)$$

Para cada imagem segmentada por uma CNN são calculadas as métricas coeficiente Dice, acurácia, sensibilidade e especificidade. O cálculo dessas métricas em um conjunto de imagens é feito por meio da média dos valores obtidos para cada imagem individualmente.



# Capítulo 3

## Revisão de Literatura

Neste capítulo são apresentadas as principais publicações sobre técnicas de *machine learning* e *deep learning* aplicadas em tarefas relacionadas à identificação de lesões de esclerose múltipla. Algumas destas pesquisas têm aplicações em outras áreas, mas utilizam métodos que foram relevantes para o desenvolvimento deste trabalho.

Valverde *et al.* [28] propõem um método para segmentação de lesões de esclerose múltipla, na massa branca do cérebro, utilizando duas CNNs tridimensionais cascadeadas. O modelo utiliza duas bases de dados: a base *Medical Image Computing and Computer Assisted Intervention (MICCAI) 2008 MS Lesion Segmentation*, que possui exames de ressonância magnética de 45 pacientes, e a *Clinical MS Dataset*, composta por exames de 60 pacientes. Ambos os conjuntos possuem poucas imagens. Para minimizar esse problema é aplicada a técnica de aumento de dados, a fim de aumentar o número de imagens e melhorar o desempenho do modelo. A arquitetura proposta obteve ótimos resultados e exibiu um treinamento consistente quando aplicada em diferentes imagens.

Seguindo uma abordagem semelhante, Afzal *et al.* [29] aplicam duas CNNs bidimensionais cascadeadas para segmentar lesões de esclerose múltipla com maior precisão. A primeira realiza a tarefa de segmentação e a segunda refina os resultados, reduzindo o número de falsos positivos para aumentar a eficiência. O modelo é aplicado em duas bases de dados, adquiridas por meio das convenções MICCAI e *International Symposium on Biomedical Imaging (ISBI)*, e compostas pelas sequências de ressonância magnética FLAIR, T1 e T2. Os resultados são consistentes e alcançaram uma precisão de 90% e coeficiente Dice de 67%. Entretanto, o algoritmo apresenta limitações na segmentação de lesões muito próximas, mostrando-se incapaz de identificá-las com alta precisão.

La Rosa *et al.* [9] desenvolveram um *framework* para a segmentação de lesões de esclerose múltipla corticais e lesões na massa branca do cérebro aplicando uma arquitetura UNet tridimensional. Nesse experimento são utilizadas 3856 imagens de lesões na massa branca e 728 imagens de lesões corticais, obtidas a partir das sequências de res-

sonância magnética FLAIR e *Magnetization-Prepared 2 Rapid Acquisition Gradient Echo* (MP2RAGE). Visto que as imagens de lesões corticais são escassas, os autores adotam a técnica de aumento de dados, aplicando rotações de 90 graus e giros aleatórios nos dados. O modelo proposto obteve ótimos resultados, com um coeficiente Dice de 62%. Contudo, as lesões corticais são mais difíceis de segmentar devido ao pouco contraste com o restante da imagem.

Outra abordagem encontrada na literatura é o uso de modelos de CNN *Attention-guided*. Uma *Attention-guided Two-Path Convolutional Neural Network* (TPCNN) é proposta pelos autores Gessert *et al.* [30]. Eles utilizam dois exames de ressonância magnética dos mesmos pacientes, obtidos em um intervalo de 1 a 3 anos, aproximadamente, com o objetivo de realizar a segmentação de lesões de esclerose múltipla e analisar a progressão da doença comparando-se os resultados. Para a execução dessa tarefa, é utilizada uma base de dados composta por imagens de ressonância magnética FLAIR de 89 pacientes. O método proposto é também comparado com os métodos *Lesion Segmentation Tool* (LST) *toolbox* e uma arquitetura UNet. A TPCNN obteve um coeficiente Dice de 56,77%.

Hashemi *et al.* [14] publicaram um trabalho que aplica uma *Attention* UNet modificada para a segmentação de lesões de esclerose múltipla em imagens de ressonância magnética FLAIR e T2. A base de dados utilizada nessa pesquisa é a ISBI 2015 *Challenge Dataset*, composta por 20 exames de 5 pacientes. A primeira etapa da metodologia proposta é o pré-processamento dos dados, cujo objetivo é deletar imagens sem lesões e diminuir o tempo de processamento. Na etapa seguinte, o modelo proposto é comparado à uma arquitetura UNet modificada. A *Attention* UNet modificada obteve melhores resultados nos experimentos, alcançando um coeficiente Dice de 82,30%.

Acar *et al.* [31] apresentam um método para a identificação de lesões de esclerose múltipla em imagens de ressonância magnética FLAIR aplicando uma CNN. Os autores propõem uma arquitetura com poucas camadas e, conseqüentemente, um baixo número de parâmetros de treinamento. São aplicadas 1942 imagens utilizando a técnica de validação cruzada com 5 *folds* e dividindo as imagens por paciente e por *slice*. Por meio da divisão de dados por paciente, foram obtidas acurácia de 90,3%, revocação de 90,5% e, especificidade de 90,1%; a divisão de imagens por *slice* gerou uma acurácia de 98,0%, revocação de 97,9% e especificidade de 98,3%. Os resultados são comparados com os algoritmos de classificação *Support Vector Machine* (SVM), *Logistic Regression* (LR), *K-Nearest Neighbors* (KNN) e *Decision Tree* (DT), e com as arquiteturas VGG16, VGG19, ResNet50, ResNet101, e InceptionV3.

CNNs são muito eficazes na tarefa de segmentação, mas elas podem ter um baixo desempenho quando são expostas a uma base de dados desconhecida. A fim de contornar essa questão, Kamraoui *et al.* [32] desenvolveram uma nova técnica generalizada de seg-

mentação de lesões de esclerose múltipla, capaz de obter bons resultados mesmo quando aplicada em novas imagens. Essa técnica, conhecida como *Deep Lesion Brain*, consiste em várias CNNs tridimensionais compactadas, sobre as quais é aplicada a técnica de aprendizado *Hierarchical Specialization Learning* (HSL) e a técnica de aumento de dados. O novo modelo foi aplicado em três bases de dados e mostrou ser muito eficiente na generalização das imagens.

Os autores Gaj *et al.* [33] implementam uma UNet bidimensional para segmentar lesões de esclerose múltipla realçadas por gadolínio e, em seguida, aplicam um classificador *Random Forest* para contar o número de lesões. O método é aplicado em duas bases de dados compostas por 600 e 2846 imagens, respectivamente, e sequências de ressonância magnética FLAIR, T1, T2 e *Proton Density* (PD). Foi obtida uma revocação de 96,9% e um coeficiente Dice de 76,7%.

A técnica de *transfer learning* tem como objetivo melhorar o desempenho de uma CNN quando a base de dados não possui imagens suficientes. A partir disso, Cheng e Lam [13] realizaram um estudo sobre qual técnica de *transfer learning* e *fine-tuning* seria mais eficiente em uma arquitetura UNet para a segmentação de costelas em imagens de ultrassom do pulmão. Neste estudo são consideradas duas abordagens: o uso de um modelo VGG16 pré-treinado na construção da arquitetura da UNet e o pré-treinamento da UNet em uma base de dados mais robusta e, posteriormente, afinada com as imagens de ultrassom. A técnica de aumento de dados também é aplicada sobre a base de dados, dobrando o número de imagens de 200 para 400. A VGG16 UNet com *fine-tuning* obteve os melhores resultados, com um coeficiente Dice de 86,32%.

Essa *et al.* [34] apresentam um método automatizado para segmentação de lesões de esclerose múltipla, em imagens FLAIR e T2, baseado em uma *Region-Based Convolutional Neural Network* (R-CNN) com *transfer learning*. Para melhorar os resultados é realizado o pré-processamento das imagens, etapa onde é aplicada a técnica de *skull-stripping*, utilizada para remover o crânio e outros tecidos não relevantes do cérebro, e *bias field correction*, destinada à remoção de ruídos da imagem. Em seguida, a R-CNN realiza a segmentação das duas sequências de ressonância magnética e, por fim, une os produtos a partir do algoritmo *Adaptive Neuro-Fuzzy Inference System* (ANFIS). A aplicação do ANFIS reduz a quantidade de falsos positivos e, conseqüentemente, gera melhores resultados do que o modelo individual da R-CNN.

Uma técnica que tem sido muito explorada na tarefa de segmentação é a combinação das arquiteturas ResNet e UNet para a construção de um modelo híbrido chamado ResUNet. Essa metodologia é proposta por Rahman *et al.* [35] para realizar a segmentação do fígado e de tumores no fígado em imagens de tomografia. Também é aplicada a técnica de aumento de dados para aumentar o número de imagens disponível. Apesar de algumas

limitações, como o tamanho da base de dados, os resultados se mostraram promissores, com uma acurácia de 99,6% e um coeficiente Dice de 99,2%.

As obras mencionadas anteriormente utilizaram técnicas baseadas em CNNs em seus estudos. Contudo, há pesquisas na literatura que aplicam métodos tradicionais de segmentação, como o trabalho de Gheshlaghi *et al.* [36]. Eles publicaram um artigo com o objetivo de classificar pacientes saudáveis e pacientes com lesões de esclerose múltipla, a partir da sequência de ressonância magnética FLAIR, utilizando as técnicas SVM, *Discrete Wavelet Decomposition* (DWT) e segmentação de superpixels. Para melhorar o desempenho do modelo, é aplicada uma etapa de pré-processamento na qual é realizada a extração do cérebro das imagens utilizando a ferramenta *Brain Extraction Tool* (BET). Em seguida, é aplicada a técnica de segmentação de superpixels e DWT. Por fim, os dados são fornecidos a uma SVM que classifica as imagens. A partir da validação cruzada *k-fold*, com 10 *folds*, é obtida uma acurácia de 99,65%.

Cetin *et al.* [37] propõem um novo método de segmentação e classificação, baseado no cálculo de distância mínima euclidiana, para realizar a análise quantitativa de lesões de esclerose múltipla. O modelo de aprendizado supervisionado é utilizado para classificar os principais tecidos do cérebro em 6 categorias: massa cinzenta, massa branca, líquido cefalorraquidiano, osso, pele e lesões de esclerose múltipla. O novo método é aplicado em uma base de dados com imagens de ressonância magnética de 6 pacientes, composta pelas sequências FLAIR, T1 e T2. Esse conjunto de imagens é dividido a nível de paciente por meio das técnicas de validação cruzada com 2 *folds* e validação cruzada *Leave One Out*.

A Tabela 3.1 exibe uma síntese de cada um dos trabalhos citados neste capítulo, incluindo técnicas utilizadas e resultados obtidos. Detalhes sobre como são calculadas as métricas mencionadas nesta seção podem ser encontradas na Seção 2.8.

Por meio da análise dos dados da Tabela 3.1, é possível notar que a CNN é a técnica predominante nesses trabalhos e que a UNet é a arquitetura mais explorada na segmentação de lesões de esclerose múltipla. Os modelos ResUNet e VGG16-UNet<sup>1</sup> obtêm bons resultados, mas não é possível compará-los aos outros estudos pois são empregados na resolução de problemas diferentes. Este trabalho propõe o uso das CNNs UNet, VGG16-UNet, SegNet e ResUNet na segmentação de lesões de esclerose múltipla. Dessa forma, é possível verificar a eficácia das variações ResUNet e VGG16-UNet para o problema e comparar os resultados com a UNet, uma rede amplamente utilizada. A SegNet é um modelo cuja arquitetura difere significativamente das outras redes propostas. Assim, é interessante analisar seu desempenho juntamente com as outras CNNs. O *transfer learning* e o aumento de dados, aplicados em alguns dos trabalhos citados anteriormente, geralmente têm um impacto positivo nos resultados e, devido a isso, também são utilizados.

---

<sup>1</sup>Variação da arquitetura UNet, com aplicação de *transfer learning*

Tabela 3.1: Resumo dos trabalhos apresentados neste capítulo.

Ano	Autores	Problemas	Técnicas	Resultados
2017	Valverde <i>et al.</i> [28]	Segmentação de lesões de esclerose múltipla	CNNs tridimensionais e aumento de dados	Dice: 56%
2021	Afzal <i>et al.</i> [29]	Segmentação de lesões de esclerose múltipla	CNNs bidimensionais	Dice: 67%
2020	La Rosa <i>et al.</i> [9]	Segmentação de lesões de esclerose múltipla corticais e na massa branca do cérebro	UNet tridimensional e aumento de dados	Dice: 62%
2020	Gessert <i>et al.</i> [30]	Segmentação de lesões de esclerose múltipla	<i>Attention-guided</i> TPCNN	Dice: 56,77%
2022	Hashemi <i>et al.</i> [14]	Segmentação de lesões de esclerose múltipla	<i>Attention</i> UNet modificada	Dice: 82,30%
2022	Acar <i>et al.</i> [31]	Classificação de lesões de esclerose múltipla	CNN	Acurácia: 98,0% Revocação: 97,9% Especificidade: 98,3%
2022	Kamraoui <i>et al.</i> [32]	Segmentação de lesões de esclerose múltipla	CNN tridimensional, HSL e aumento de dados	Dice: 74,6%
2021	Gaj <i>et al.</i> [33]	Segmentação e classificação de lesões de esclerose múltipla realçadas por gadolínio	UNet bidimensional e <i>Random Forest</i>	Dice: 76,7% Revocação: 96,9%
2021	Cheng e Lam [13]	Segmentação de costelas em imagens de ultrassom do pulmão	Unet, <i>transfer learning</i> e aumento de dados	Dice: 86,32%
2020	Essa <i>et al.</i> [34]	Segmentação de lesões de esclerose múltipla	R-CNN, <i>transfer learning</i> e ANFIS	Revocação: 61,8%
2022	Rahman <i>et al.</i> [35]	Segmentação do fígado e de tumores no fígado	ResUNet e aumento de dados	Dice: 99,2% Acurácia: 99,6%
2019	Gheshlaghi <i>et al.</i> [36]	Classificação de lesões de esclerose múltipla	SVM, DWT e segmentação de superpixels	Acurácia: 99,65%
2020	Cetin <i>et al.</i> [37]	Segmentação de lesões de esclerose múltipla e classificação de tecidos do cérebro	Classificação por distância mínima euclidiana	Dice: 71% Revocação: 65%

# Capítulo 4

## Metodologia

Neste capítulo é apresentada a metodologia aplicada neste trabalho. Ela consiste no treinamento de diferentes arquiteturas de CNN utilizando uma base de dados composta por diversas sequências de ressonância magnética e na análise e comparação desses resultados. Ademais, a técnica de aumento de dados é aplicada em cada experimento, a fim de avaliar seu impacto no desempenho dos modelos. A Figura 4.1 representa graficamente a metodologia empregada.

Na Seção 4.1 são mostrados detalhes sobre a base de dados e suas imagens; a Seção 4.2 aborda as principais técnicas aplicadas nas imagens antes do treinamento, com o intuito de melhorar o desempenho dos modelos; a Seção 4.3 detalha as arquiteturas utilizadas em cada experimento; por fim, a Seção 4.4 explica o processo de análise e comparação dos experimentos.

### 4.1 Base de dados

Para a realização dos procedimentos experimentais deste capítulo foi empregada a base de dados do desafio de segmentação de lesões de esclerose múltipla longitudinais, organizado pelo ISBI em 2015<sup>1</sup>. O desafio forneceu aos participantes três conjuntos de imagens: um conjunto para treinamento e dois para teste, chamados A e B, respectivamente. A base com as imagens de treinamento foi disponibilizada para o público e consiste em exames de ressonância magnética de cinco pacientes no total, sendo quatro mulheres e um homem. Quatro dos pacientes possuem quatro exames, enquanto o quinto possui cinco. Cada ressonância foi adquirida em um intervalo de tempo de aproximadamente um ano em relação ao exame subsequente. Dentre os cinco pacientes, quatro têm esclerose múltipla recorrente-remitente; o quinto está no estágio primariamente progressivo [38].

---

<sup>1</sup><https://biomedicalimaging.org/2015/program/isbi-challenges/>

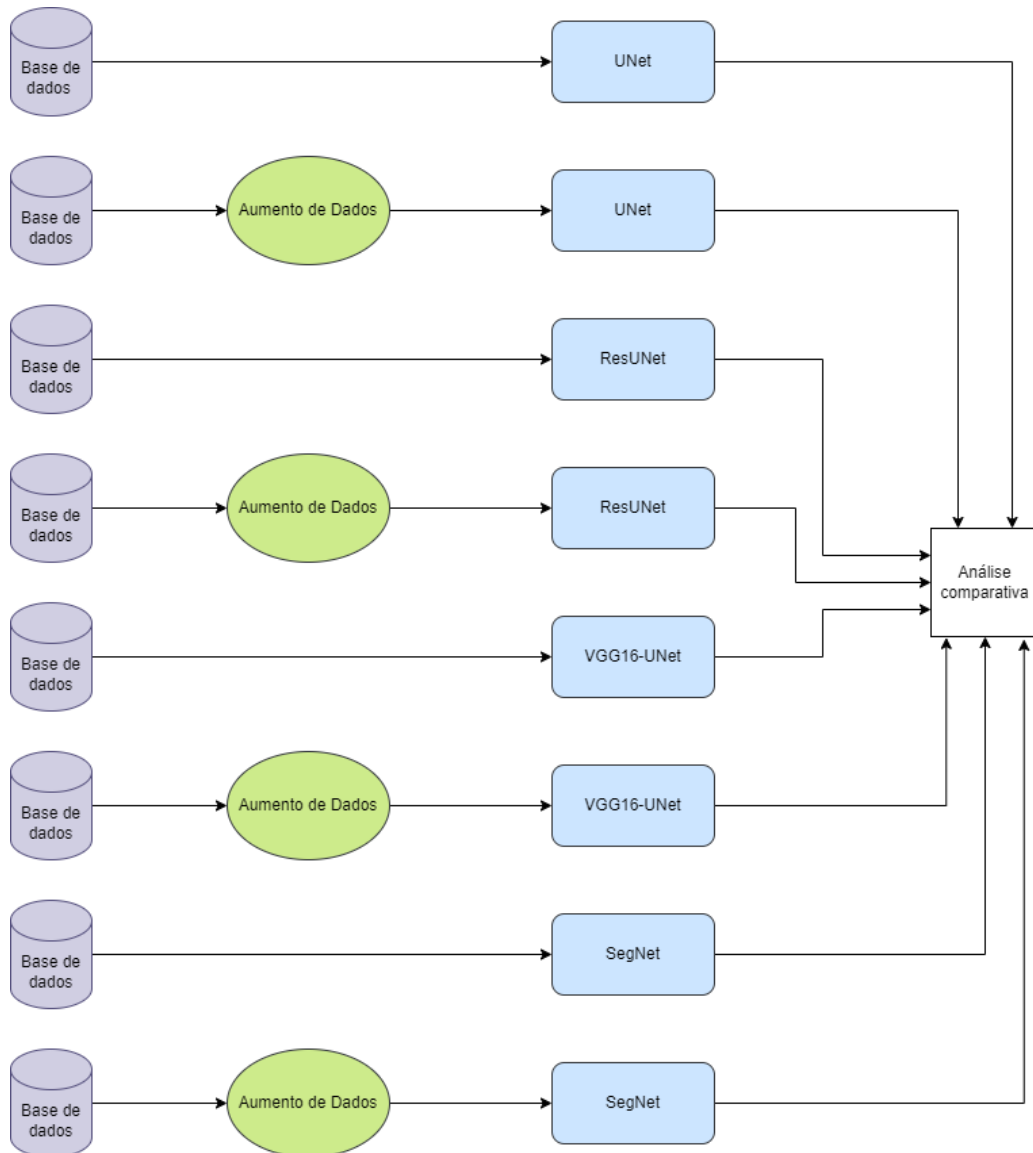


Figura 4.1: Diagrama da metodologia proposta.

As ressonâncias foram capturadas por um profissional da área da saúde em um *scanner* 3.0 Tesla usando as sequências T2, FLAIR, PD e MP-RAGE. Cada aquisição passou pelas seguintes etapas de pré-processamento: correção de viés por meio do algoritmo N4 [39]; remoção do crânio e da dura-máter por meio de algoritmos de *skull-stripping* [40] e *dura-stripping* [41], respectivamente; e, por último, o registro em um *template MNI*<sup>2</sup> de 1 milímetro. Essas etapas são importantes na redução de ruídos que surgem durante a captura das imagens e na sua padronização. Na Figura 4.2 é possível visualizar o resultado do pré-processamento. Todas as imagens foram analisadas por dois profissionais especialistas em detecção de lesões no cérebro e ambos identificaram as lesões manualmente. Por fim, a base de dados disponibilizada para o desafio é formada pelas imagens originais

<sup>2</sup><https://www.lead-dbs.org/about-the-mni-spaces/>

T2, FLAIR, PD e MP-RAGE, as imagens pré-processadas correspondentes e as imagens das lesões manualmente identificadas pelos dois especialistas, também conhecidas como *ground-truth*.

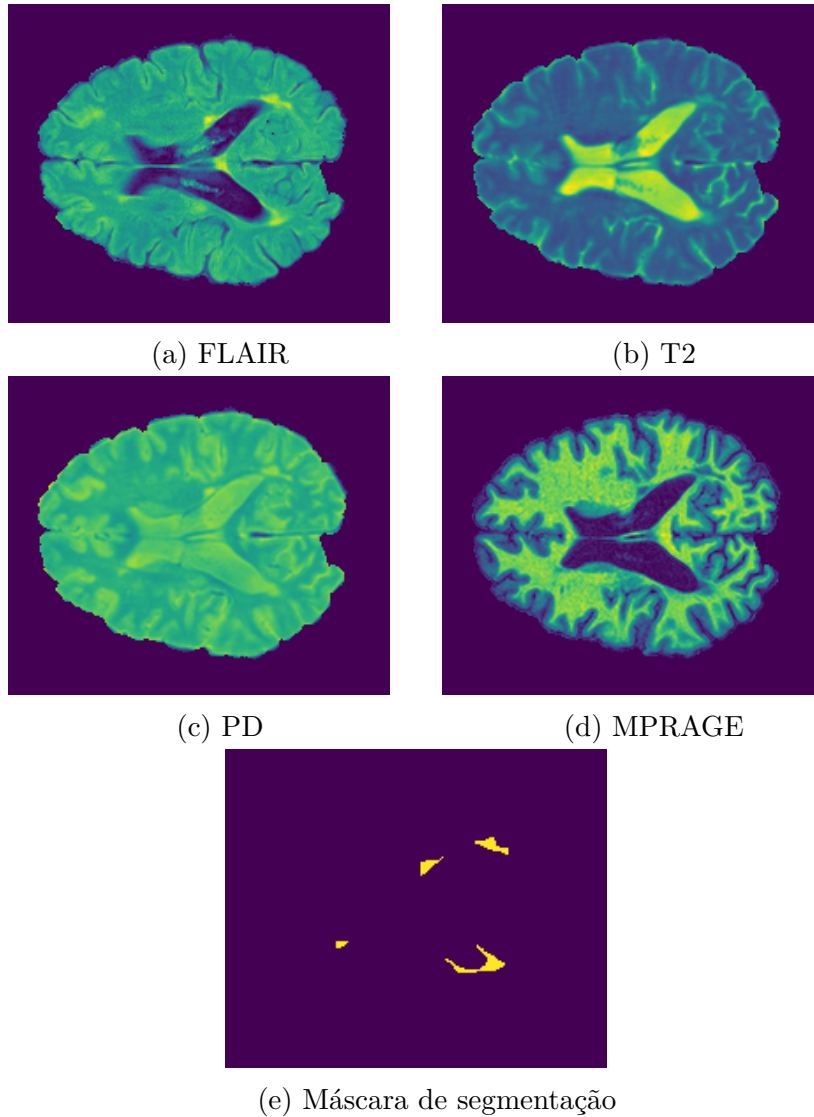


Figura 4.2: Exemplos de imagens pré-processadas da base de dados para cada sequência de ressonância magnética [2].

Cada ressonância é armazenada em um arquivo NIfTI, resultando em um total de 84 arquivos. Um arquivo contém 181 *slices*, que são convertidos em imagens PNG neste trabalho e geram 3801 imagens para cada sequência. Somando o número de imagens de cada sequência de ressonância magnética, obtém-se 15204 imagens, com e sem lesões, de dimensões  $217 \times 181$  *pixels*. Como parte do objetivo deste trabalho é a segmentação de lesões de esclerose múltipla, imagens sem lesões não são relevantes para esta tarefa. Após a exclusão desses itens, restam 1392 imagens para cada sequência. Os experimentos com as redes neurais exigem três conjuntos de dados para as seguintes tarefas: treinamento,



validação e teste do modelo. Assim, foram destinadas 70% das imagens para treinamento, 10% para validação e 20% para teste, distribuídas de forma aleatória. Essas porcentagens equivalem a 1001, 112 e 279 imagens, respectivamente.

## 4.2 Pré-processamento das imagens

Antes de alimentar as imagens a um modelo de CNN, é necessário padronizá-las para que sejam aceitas como dados de entrada pela rede neural e para aumentar o desempenho durante o treinamento. As etapas de pré-processamento aplicadas nos experimentos deste capítulo são: redimensionamento, normalização de intensidades e aumento de dados.

### 4.2.1 Redimensionamento

As imagens da base de dados, mencionadas na Seção 4.1, possuem dimensões  $217 \times 181$  *pixels*. Com o fim de diminuir o custo computacional durante o treinamento das CNNs, todas as imagens são redimensionadas para um tamanho reduzido de  $160 \times 160$  *pixels*, sem alterar o sistema de cores RGB. Primeiramente, a altura da imagem é cortada, para que seja obtida uma imagem quadrada de  $181 \times 181$  *pixels*. Nenhuma região de interesse é afetada nesse processo. Em seguida é feito o redimensionamento, utilizando uma interpolação bilinear.

Esse processo é realizado com o intuito de diminuir ao máximo a perda de qualidade da imagem. É recomendável que as dimensões dos dados de entrada das redes utilizadas neste trabalho sejam múltiplas de 32. Dessa forma, foi calculado, de forma experimental, que o redimensionamento das imagens em  $160 \times 160$  *pixels* resultou na menor perda de qualidade.

### 4.2.2 Normalização de intensidades

A normalização de imagens é a segunda técnica aplicada durante o pré-processamento. Esse é um método comum na área de aprendizado de máquina e consiste em alterar o intervalo que a intensidade de um *pixel* pode assumir. Em todos os experimentos, as imagens, originalmente com intensidades entre 0 e 255, são normalizadas no intervalo entre 0 e 1. As lesões manualmente delineadas, ou as imagens de *ground-truth*, são binarizadas em seguida e seus *pixels* passam a assumir os valores 0 e 1, equivalentes as cores preto e branco, respectivamente.

A normalização é uma etapa importante pois pode afetar o comportamento de uma CNN. Na Seção 2.4 é explicado que os neurônios da rede neural geram um valor de saída a partir de cálculos aplicados sobre o valor do *pixel* de entrada. Esses cálculos incluem a

multiplicação do *pixel* pelos pesos do neurônio e a adição do viés. A normalização mantém esses valores consistentes e impede que eles aumentem muito e desacelerem o treinamento.

### 4.2.3 Aumento de dados

A última etapa do pré-processamento é o aumento de dados. Esse método, citado na Seção 2.7, gera novas imagens a partir dos dados originais por meio de pequenas alterações em suas características. O aumento de dados é aplicado sobre o conjunto de imagens de treinamento (1001) e validação (112), antes do treinamento da CNN, e aumenta o número de artefatos de ambos os conjuntos em 50%, gerando 500 novas imagens para treinamento e 56 imagens para validação. Esses valores são escolhidos de forma empírica. As transformações executadas<sup>3</sup> [10] são citadas na lista a seguir e representadas na Figura 4.3. Elas são frequentemente utilizadas nos trabalhos citados no Capítulo 3.

- Rotações aleatórias da imagem em até 20 graus;
- *Zoom* aleatório de até 20% em relação ao valor original da imagem;
- Giros aleatórios da imagem na vertical e horizontal.

## 4.3 Experimentos propostos

Para realizar a segmentação de lesões de esclerose múltipla, são utilizadas as imagens descritas na Seção 4.1, em conjunto com o *ground-truth* fornecido pelo primeiro especialista, escolhido de forma arbitrária. Ademais, são empregadas as CNNs UNet [24], ResUNet [27], VGG16-UNet [13] e SegNet [26]. Cada arquitetura é treinada com uma sequência de ressonância magnética individualmente. A Figura 4.4 exemplifica os experimentos propostos para cada sequência. São realizados dois experimentos principais: o treinamento dos modelos com a aplicação do aumento de dados, durante a etapa de pré-processamento, e sem ele, a fim de avaliar seu impacto nos resultados. Os procedimentos de redimensionamento e a normalização são utilizados em todos os experimentos.

CNNs e algoritmos de aprendizado de máquina possuem algumas configurações específicas, como o otimizador e a função de perda. O otimizador é um algoritmo que ajusta os pesos e a taxa de aprendizado, diminuindo a perda de informações durante o treinamento. Nos experimentos é utilizado o otimizador Adam com a taxa de aprendizado padrão de 0.001. A função de perda adotada é a *Binary Cross-entropy*, comumente aplicada em classificações binárias. Ela retorna valores baixos para boas previsões e valores altos para previsões ruins do modelo.

---

<sup>3</sup>Não é possível manter as proporções da imagem para todas as transformações.

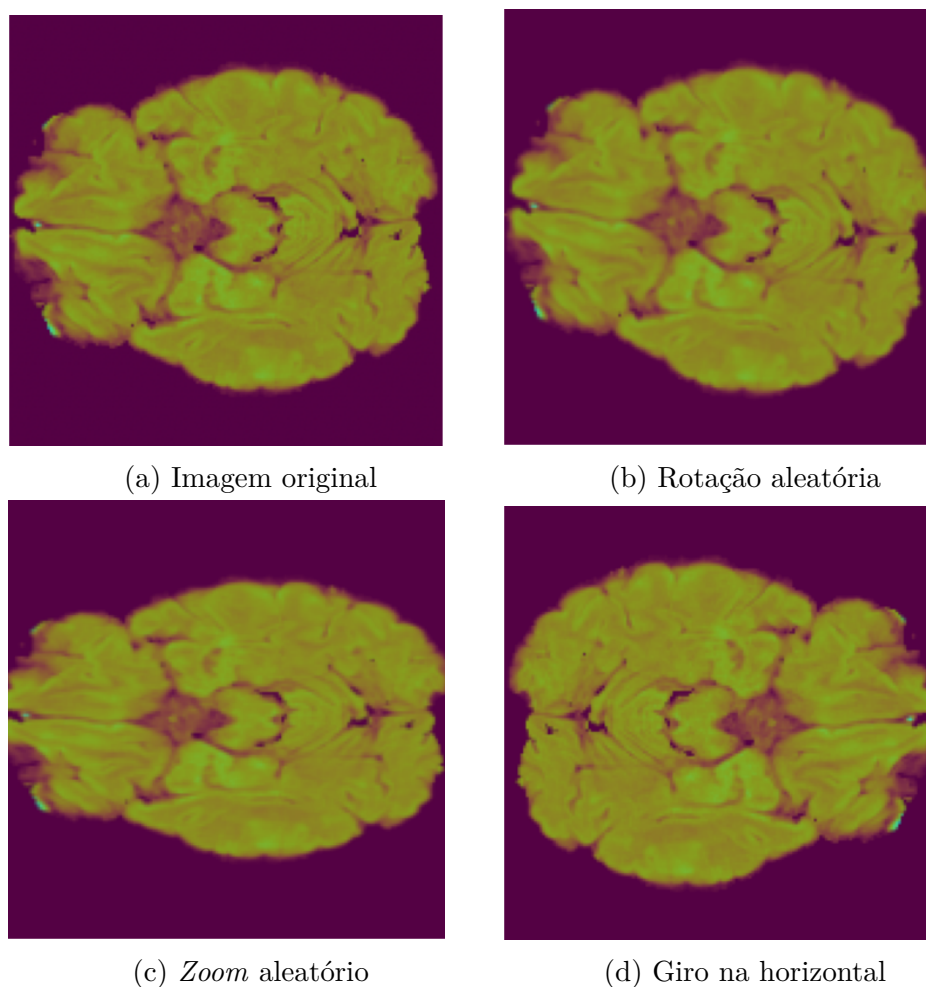


Figura 4.3: Exemplos de imagens geradas após a aplicação da técnica de aumento de dados.

Durante o treinamento, é possível que os valores correspondentes as métricas parem de crescer após algum tempo devido ao *overfitting* do modelo. Neste trabalho, as CNNs são configuradas para treinar durante 30 épocas, porém é adicionada uma verificação extra conhecida como parada antecipada, que encerra o treinamento assim que o valor do coeficiente Dice deixa de aumentar. A parada antecipada é uma técnica de regularização que ajuda a prevenir o *overfitting*. Ela avalia o desempenho do modelo ao fim de cada época. Caso a métrica coeficiente Dice não seja superior ao valor obtido na época anterior, o treinamento é encerrado. Essa técnica é aplicada a partir de 15 épocas.

A partir dessas configurações são implementados os modelos da UNet, ResUNet, VGG16-UNet e SegNet. Detalhes sobre essas arquiteturas são explorados nas subseções a seguir.

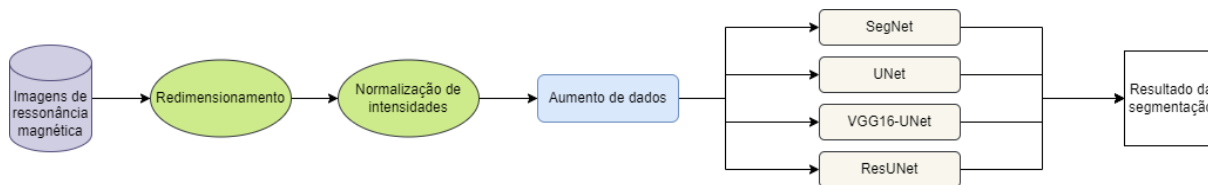


Figura 4.4: Fluxograma dos experimentos realizados para cada tipo de imagem.

### 4.3.1 UNet

A UNet é uma arquitetura frequentemente utilizada na tarefa de segmentação semântica de imagens médicas, como é mencionado na Subseção 2.5.1. Um dos motivos pelos quais esse modelo se popularizou ao longo dos anos são as conexões de salto, também conhecidas como *skip connections*. À medida que as informações da imagem são passadas de uma camada para outra, é possível que alguns dados importantes sejam perdidos. As *skip connections* permitem que essas informações sejam recuperadas por meio da introdução do mapa de características, obtido no caminho de contração, no bloco correspondente no caminho de expansão. Essa operação é capaz de aumentar a acurácia do modelo consideravelmente. O objetivo deste experimento é avaliar o desempenho da UNet, considerada um método estado da arte, na tarefa de segmentação de lesões de esclerose múltipla, para posterior comparação com outros modelos.

A arquitetura não sofreu alterações significativas em relação à proposta original de Ronneberger *et al.* [24], com exceção dos filtros das camadas convolucionais, que foram reduzidos para as dimensões 16, 32, 64, 128 e 256. Essa modificação diminuiu o número de parâmetros e o tempo de treinamento.

### 4.3.2 VGG16-UNet

A VGG16-UNet é uma variação da arquitetura UNet que explora a técnica de *transfer learning*, mencionada com detalhes na Subseção 2.5.2. Devido ao reaproveitamento das camadas pré-treinadas da CNN VGG16, a VGG16-UNet utiliza filtros com dimensões 64, 128, 256 e 512 e, conseqüentemente, contém mais parâmetros e tem um tempo de treinamento mais extenso. No entanto, a aplicação do *transfer learning* auxilia no desempenho do modelo e possibilita obter uma acurácia maior durante o treinamento. O objetivo deste experimento é verificar o impacto dessa técnica no desempenho da UNet.

### 4.3.3 SegNet

A SegNet é uma CNN simétrica que usa as camadas pré-treinadas da VGG16 no caminho de contração, característica que é detalhada na Subseção 2.5.3. Diferentemente da UNet, a

SegNet não contém *skip connections*. As operações de *upsampling*, responsáveis por dobrar o valor das dimensões de uma imagem, utilizam os índices das operações de *max-pooling* correspondentes para executar essa tarefa. Isso reduz o número de parâmetros do modelo e exige menos memória durante o treinamento, mas produz mapas de características mais esparsos. O objetivo deste experimento é verificar se o desempenho dessa arquitetura é superior ao da UNet.

#### 4.3.4 ResUNet

A ResUNet é uma combinação da arquitetura UNet com blocos residuais de aprendizado profundo, característica marcante da CNN ResNet. Essas unidades auxiliam no treinamento do modelo e diminuem a perda de informações durante o fluxo de camadas. Assim como a UNet, a ResUNet contém *skip connections* no caminho de expansão e dentro dos próprios blocos residuais, facilitando a propagação dos dados e diminuindo a degradação da imagem, como é citado na Subseção 2.5.4. O modelo foi desenvolvido para obter um alto desempenho durante o treinamento e foi proposto como uma melhoria da arquitetura UNet. Esse experimento tem como objetivo verificar se o modelo é mais eficiente na tarefa de segmentação do que a UNet e a SegNet. Na implementação deste trabalho, foram adicionados dois blocos de camadas extras na estrutura da ResUNet, com o intuito de aumentar a eficácia da rede. Por conseguinte, o número de parâmetros aumentou.

### 4.4 Análise comparativa

Os experimentos propostos são divididos de acordo com as arquiteturas de CNN utilizadas: UNet, VGG16-UNet, SegNet e ResUNet, como é retratado na Figura 4.1. Cada modelo é treinado nos conjuntos de imagens FLAIR, T2, MP-RAGE e PD, individualmente, com e sem a aplicação do método de aumento de dados. Em seguida, é feita uma análise comparativa dos resultados. Essa análise se divide em duas etapas: a primeira avalia o impacto da técnica de aumento de dados em cada CNN, para cada sequência de ressonância magnética; a segunda etapa é uma análise geral de todos os modelos, com o intuito de avaliar o desempenho das arquiteturas para cada conjunto de imagens. Para isso, são utilizadas as métricas coeficiente Dice, acurácia, sensibilidade e especificidade, detalhadas na Seção 2.8.

# Capítulo 5

## Resultados

Neste capítulo são apresentados os resultados dos experimentos realizados, cujos detalhes sobre a implementação são citados no Capítulo 4. A Seção 5.1 aborda as ferramentas e bibliotecas necessárias durante o desenvolvimento. As Seções 5.2, 5.3, 5.4 e 5.5 exibem os resultados da aplicação das CNNs UNet, VGG16-UNet, SegNet e ResUNet, respectivamente. Por fim, a Seção 5.6 apresenta uma síntese dos experimentos.

### 5.1 Ferramentas utilizadas

Os experimentos realizados neste trabalho foram desenvolvidos na plataforma Google Colab <sup>1</sup>. Essa ferramenta é um serviço em nuvem que permite a criação de códigos na linguagem de programação Python. Esse serviço tem como base os notebooks do Jupyter e não exige uma configuração prévia. Os recursos do Colab utilizados foram a CPU Intel(R) Xeon(R) @ 2.30GHz, 12GB de memória RAM e GPU Tesla T4 16GB. As bibliotecas em Python utilizadas foram Tensorflow<sup>2</sup>, Keras<sup>3</sup>, Nibabel, Numpy<sup>4</sup>, Matplotlib<sup>5</sup>, OpenCV<sup>6</sup> e Scikit-learn<sup>7</sup>.

### 5.2 Experimento I: UNet

Esta seção apresenta os resultados da aplicação da rede neural UNet sobre a base de dados citada na Seção 4.1. Foram realizadas duas execuções para cada sequência de

---

<sup>1</sup><https://colab.google/>

<sup>2</sup><https://www.tensorflow.org/?hl=pt-br>

<sup>3</sup><https://keras.io/>

<sup>4</sup><https://numpy.org/>

<sup>5</sup><https://matplotlib.org/>

<sup>6</sup><https://opencv.org/>

<sup>7</sup><https://scikit-learn.org/stable/>

ressonância magnética: uma com a aplicação do aumento de dados durante a etapa de pré-processamento das imagens e outra sem a aplicação desse método. O modelo foi treinado durante 30 épocas, com a configuração de parada antecipada habilitada.

Sem o uso da técnica de aumento de dados, foram empregadas 1113 imagens para treinamento (1001) e validação (112), para cada sequência de ressonância. Com o uso dessa técnica, o número de imagens de ambos os conjuntos aumentou para 1948, como é explicado na Seção 4.2.3. As Tabelas 5.1 e 5.2 apresentam os resultados encontrados no conjunto de testes (279 imagens) para os experimentos sem e com o método de aumento de dados, respectivamente. Os melhores valores para cada métrica estão realçados em negrito.

Tabela 5.1: Resultados obtidos para a arquitetura UNet sem aumento de dados.

Sequências	Coefficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.7984</b>	<b>0.9983</b>	0.8435	<b>0.9992</b>
T2	0.7450	0.9979	<b>0.8484</b>	0.9988
MP-RAGE	0.7511	0.9979	0.8244	0.9989
PD	0.7507	0.9979	0.8478	0.9988

Tabela 5.2: Resultados obtidos para a arquitetura UNet com aumento de dados.

Sequências	Coefficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.7932</b>	<b>0.9983</b>	<b>0.8639</b>	<b>0.9991</b>
T2	0.7536	0.9980	0.8171	0.9990
MP-RAGE	0.7440	0.9979	0.8185	0.9990
PD	0.7493	0.9979	0.8489	0.9988

Por meio da análise dos resultados, percebe-se que as imagens FLAIR têm um resultado mais alto para o coeficiente Dice e para a especificidade sem o uso da técnica de aumento de dados. O valor da acurácia permaneceu estável em ambos os experimentos, enquanto a sensibilidade aumentou significativamente com a aplicação dessa técnica. Dessa forma, é evidente que o experimento sem aumento de dados gera melhores resultados para a sequência FLAIR. Por outro lado, a sequência T2 apresenta valores mais altos para o coeficiente Dice, acurácia e especificidade quando é utilizado o método de aumento de dados. A única exceção é a sensibilidade, cujo valor é melhor sem o uso desse método. Assim, para essa sequência, a aplicação do aumento de dados melhora o desempenho do modelo.

Para a sequência MP-RAGE, o coeficiente Dice e a sensibilidade são melhores sem o método de aumento de dados, enquanto o valor da especificidade é mais alto com esse método. O resultado da acurácia se manteve estável em ambos os experimentos. Logo,

o experimento sem a aplicação do aumento de dados obteve melhores resultados para as imagens MP-RAGE. A sequência PD tem um resultado melhor para o coeficiente Dice sem aumento de dados e uma sensibilidade melhor com ele, enquanto as outras métricas não sofreram alterações entre os experimentos. O coeficiente Dice indica o quanto o contorno predito pelo modelo se assemelha ao contorno do *ground-truth*, sendo mais abrangente na avaliação do resultado da segmentação. A sensibilidade avalia o número de *pixels* corretamente classificados como lesões, porém não considera os *pixels* erroneamente classificados como lesões em seu cálculo. Considerando o impacto do coeficiente Dice na análise dos resultados comparado à sensibilidade, o experimento sem aumento de dados gerou valores mais altos para as imagens PD.

A Figura 5.1 mostra a curva de coeficiente Dice para cada experimento sem aumento de dados. À medida que as épocas aumentam, os pesos da rede neural são reajustados pelo otimizador e as curvas convergem. É notável que a distância entre as curvas de treinamento e validação é significativamente pequena, evidenciando que o *overfitting* do modelo é mínimo e ele possui boa capacidade de generalização quando exposto a novas imagens.

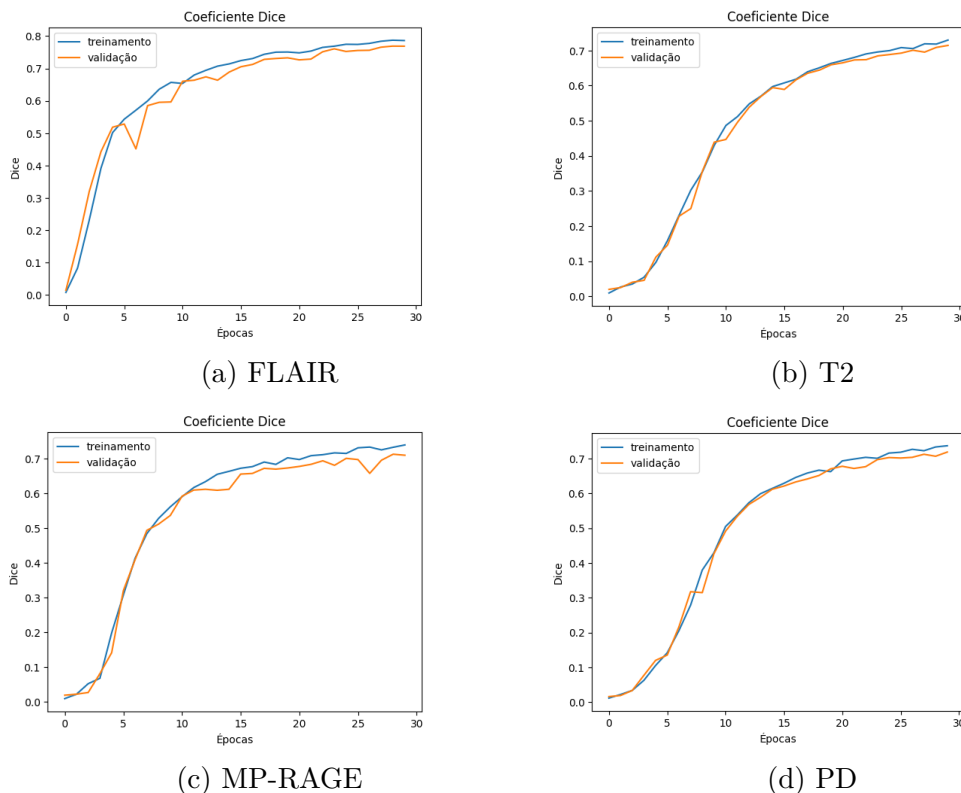


Figura 5.1: Curva de coeficiente Dice da UNet sem aumento de dados.

A Figura 5.2 exibe a curva de coeficiente Dice dos experimentos com aumento de dados. Nesse caso, é visível que houve uma divergência entre as curvas ao longo do



treinamento. Isso mostra que a UNet é capaz de boa generalização no conjunto de imagens de treinamento, porém não tem um desempenho semelhante quando é exposta a novas imagens. Esse comportamento é evidenciado pela redução nos valores dos resultados, mostrados na Tabela 5.2, sendo T2 a única exceção.

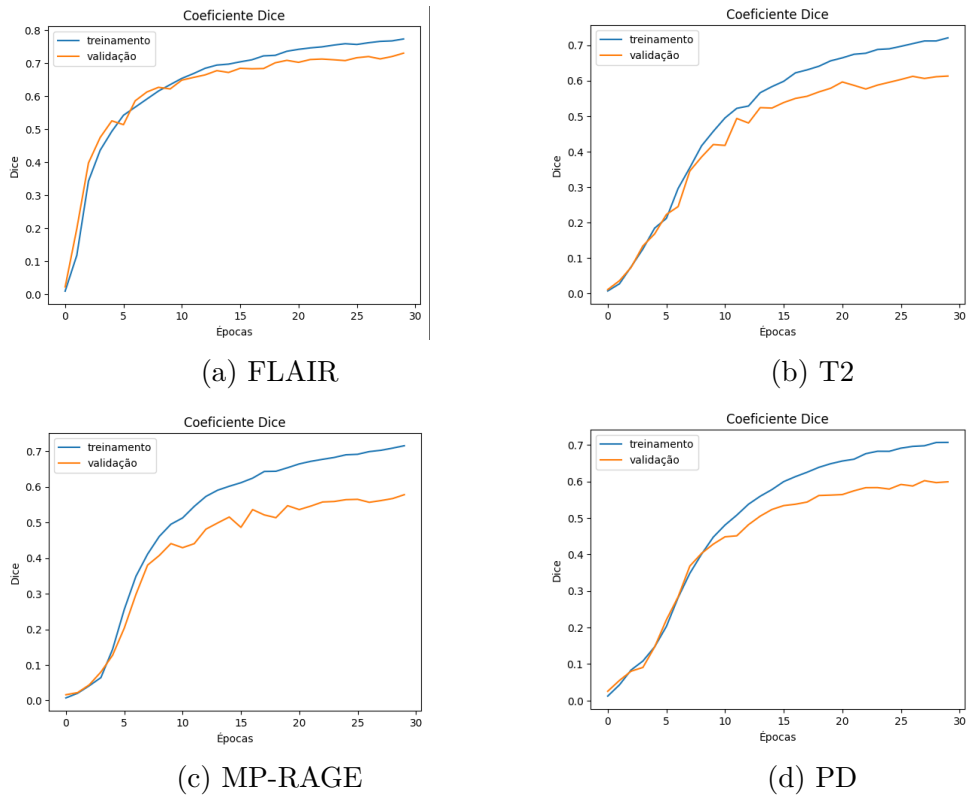
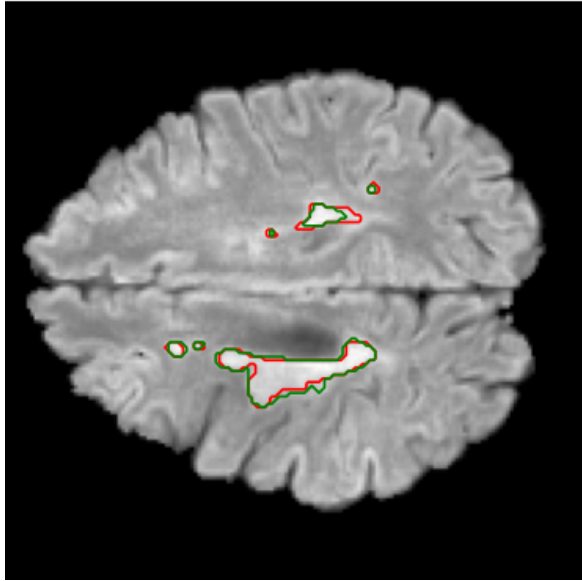
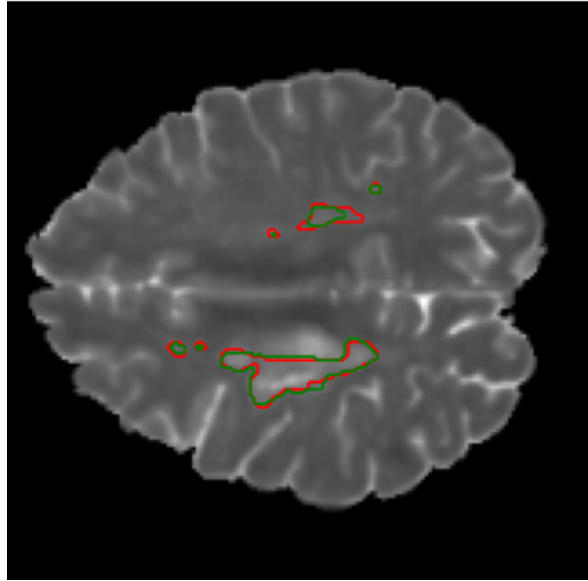


Figura 5.2: Curva de coeficiente Dice da UNet com aumento de dados.

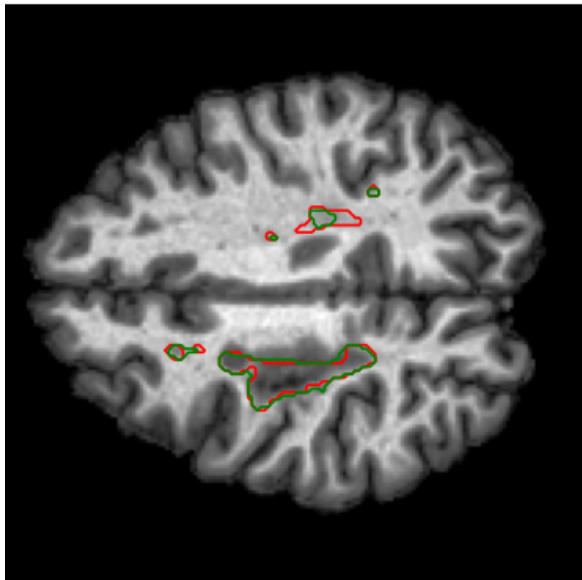
As Figuras 5.3 e 5.4 mostram as máscaras de segmentação geradas pela UNet. Foi utilizada a mesma imagem do conjunto de testes para todas as sequências e, para cada resultado, foi obtido o coeficiente Dice. Percebe-se que todas apresentam uma redução nos valores preditos quando é utilizado o aumento de dados. Diferentemente dos resultados apresentados nas Tabelas 5.1 e 5.2, o coeficiente Dice das imagens T2 também diminuiu. O resultado da UNet é muito bom para a sequência FLAIR, visto que o contorno predito é muito semelhante ao do *ground-truth*.



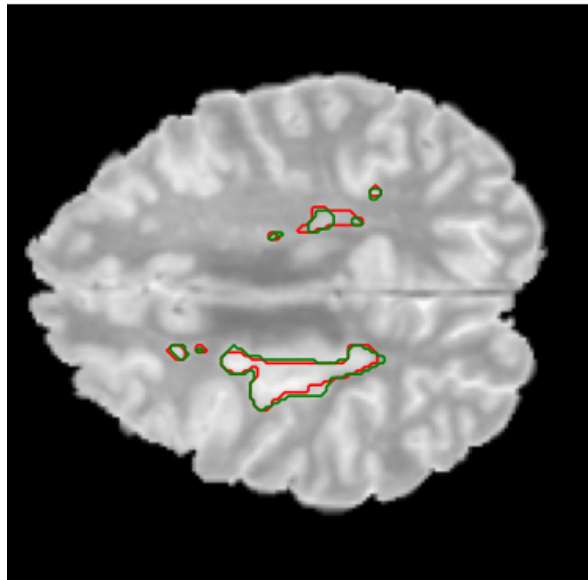
(a) FLAIR - Dice: 0.8634



(b) T2 - Dice: 0.8159

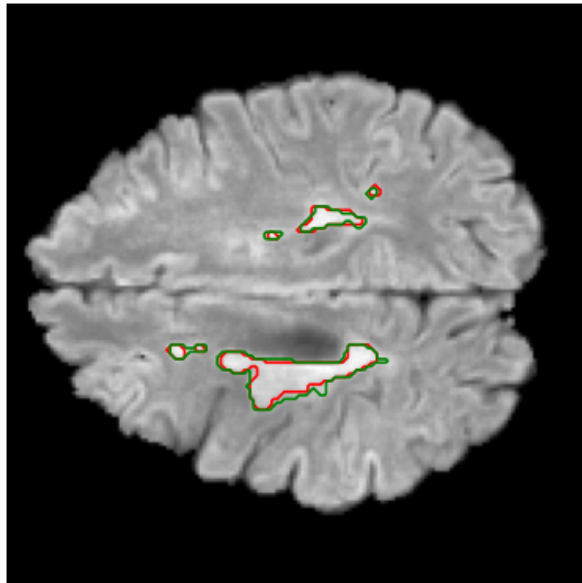


(c) MP-RAGE - Dice: 0.8244

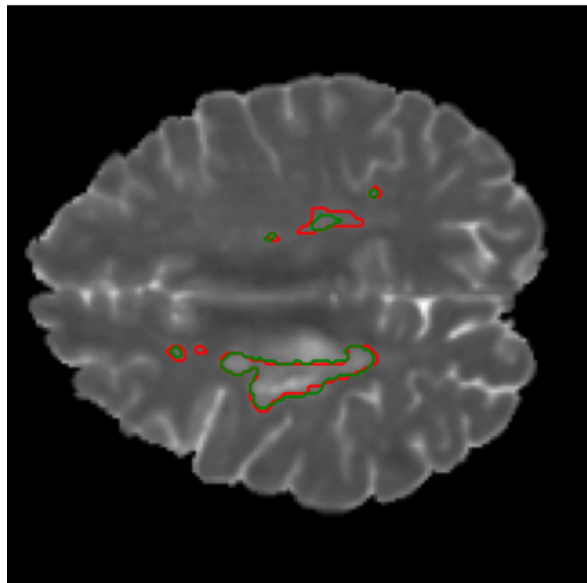


(d) PD - Dice: 0.8095

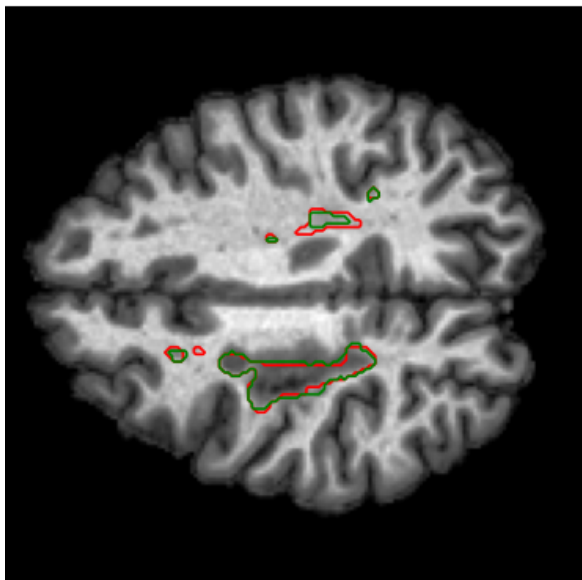
Figura 5.3: Exemplos de segmentação de lesões de esclerose múltipla com a UNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.



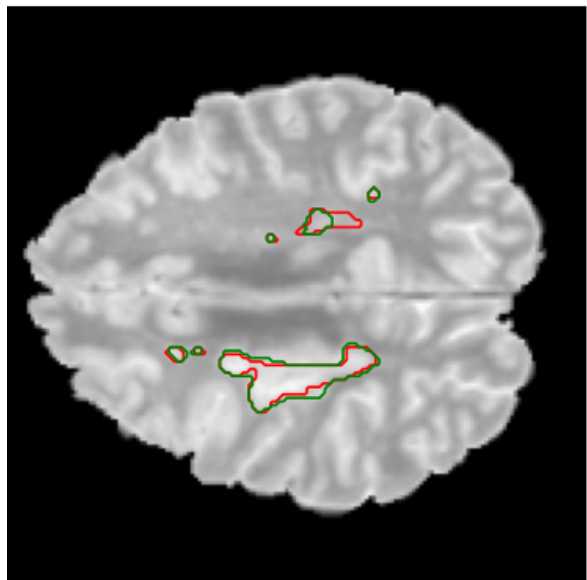
(a) FLAIR - Dice: 0.8506



(b) T2 - Dice: 0.8018



(c) MP-RAGE - Dice: 0.8195



(d) PD - Dice: 0.7987

Figura 5.4: Exemplos de segmentação de lesões de esclerose múltipla com a UNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

### 5.3 Experimento II: VGG16-UNet

Nesta seção são detalhados os resultados do uso da CNN VGG16-UNet. Essa arquitetura, citada na Subseção 2.5.2, explora a técnica de *transfer learning*. Foram realizados os mesmos experimentos da seção anterior: duas execuções para cada sequência de ressonância

magnética, com e sem o método de aumento de dados. O número de imagens utilizadas e as configurações de treinamento também foram os mesmos.

As Tabelas 5.3 e 5.4 exibem os resultados obtidos no conjunto de teste (279 imagens) para ambos os experimentos. Os melhores resultados para cada métrica estão destacados em negrito. Analisando as informações das tabelas, é notável que, para a sequência FLAIR, o coeficiente Dice e a especificidade pioraram com o uso do método de aumento de dados. A sensibilidade, no entanto, obteve um resultado melhor com o método, enquanto a acurácia se manteve constante. A sequência T2 também sofreu uma redução no valor das métricas coeficiente Dice, acurácia e especificidade com o uso do aumento de dados. A sensibilidade foi a única exceção, cujo resultado aumentou consideravelmente. Dessa forma, o experimento que alcançou melhores resultados para as imagens FLAIR e T2 foi aquele sem o aumento de dados.

Por outro lado, para a sequência de ressonância magnética MP-RAGE, o coeficiente Dice e a sensibilidade aumentaram com o uso do aumento de dados. Contudo, o resultado da especificidade sofreu uma pequena redução e o valor da acurácia não foi alterado. Já as imagens PD alcançaram resultados melhores para o coeficiente Dice e a especificidade com a técnica de aumento de dados. Porém, a sensibilidade diminuiu significativamente com a aplicação dessa técnica. A acurácia se manteve constante em ambos os experimentos. Logo, o aumento de dados melhorou o desempenho do modelo para as sequências MP-RAGE e PD.

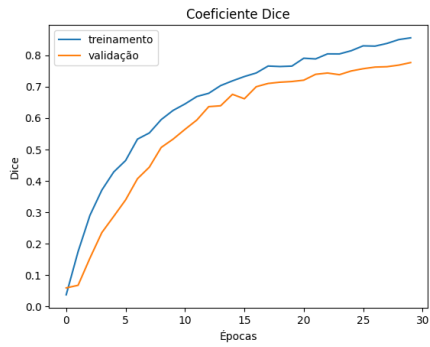
Tabela 5.3: Resultados obtidos para a arquitetura VGG16-UNet sem aumento de dados.

Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.8109</b>	<b>0.9984</b>	<b>0.8815</b>	0.9991
T2	0.7607	0.9982	0.8029	<b>0.9993</b>
MP-RAGE	0.7600	0.9981	0.7934	<b>0.9993</b>
PD	0.7438	0.9981	0.8640	0.9988

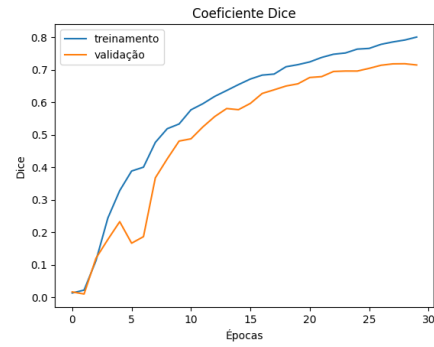
Tabela 5.4: Resultados obtidos para a arquitetura VGG16-UNet com aumento de dados.

Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.8069</b>	<b>0.9984</b>	<b>0.8928</b>	0.9990
T2	0.7416	0.9980	0.8669	0.9988
MP-RAGE	0.7706	0.9981	0.8132	0.9991
PD	0.7726	0.9981	0.7991	<b>0.9993</b>

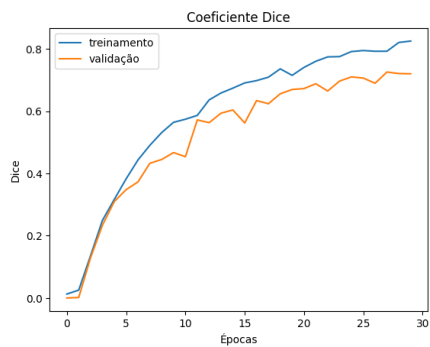
As Figuras 5.5 e 5.6 representam as curvas do coeficiente Dice para os experimentos sem e com o método de aumento de dados, respectivamente. As informações da Figura 5.6 mostram que esse método introduziu um pouco de *overfitting* no modelo, evidenciado pelo aumento na distância entre as curvas de treinamento e validação.



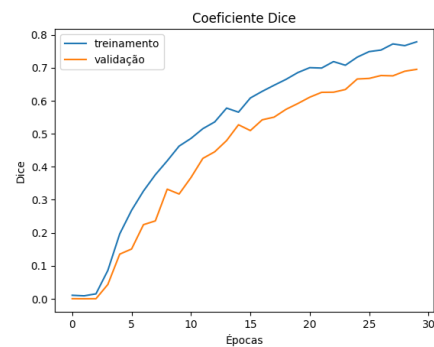
(a) FLAIR



(b) T2

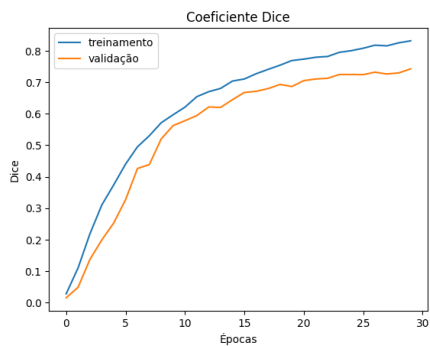


(c) MP-RAGE

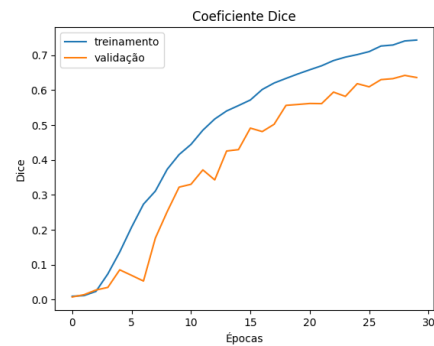


(d) PD

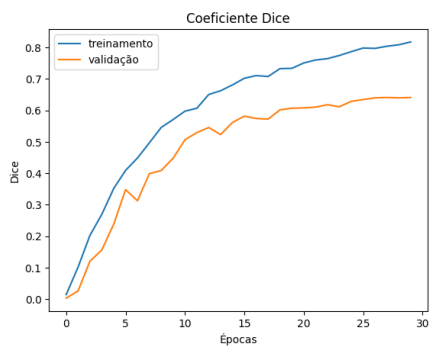
Figura 5.5: Curva de coeficiente Dice da VGG16-UNet sem aumento de dados.



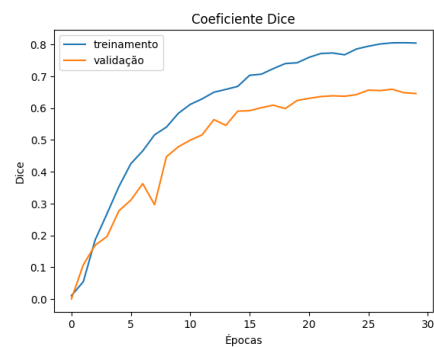
(a) FLAIR



(b) T2



(c) MP-RAGE



(d) PD

Figura 5.6: Curva de coeficiente Dice da VGG16-UNet com aumento de dados.

Por fim, as Figuras 5.7 e 5.8 exibem alguns exemplos de resultados da segmentação gerados pela VGG16-UNet. Todas as sequências alcançaram ótimos resultados para o coeficiente Dice, com valores acima de 0.8 para todas as imagens. Contudo, ao contrário das médias calculadas nas Tabelas 5.3 e 5.4, o aumento de dados reduziu o resultado da imagem MP-RAGE e melhorou o resultado da imagem FLAIR. As sequências T2 e PD seguem com o mesmo comportamento observado anteriormente.

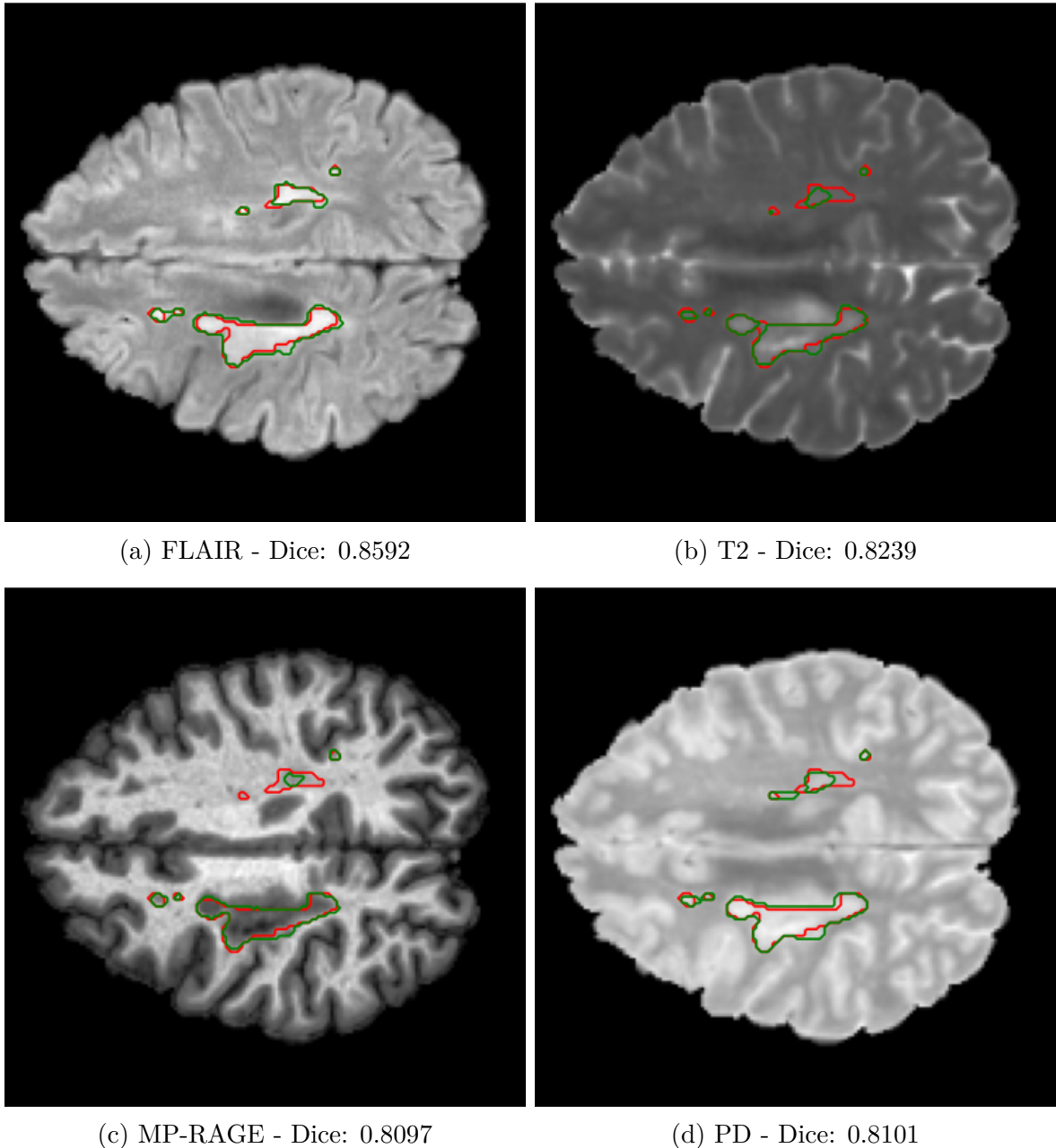


Figura 5.7: Exemplos de segmentação de lesões de esclerose múltipla com a VGG16-UNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

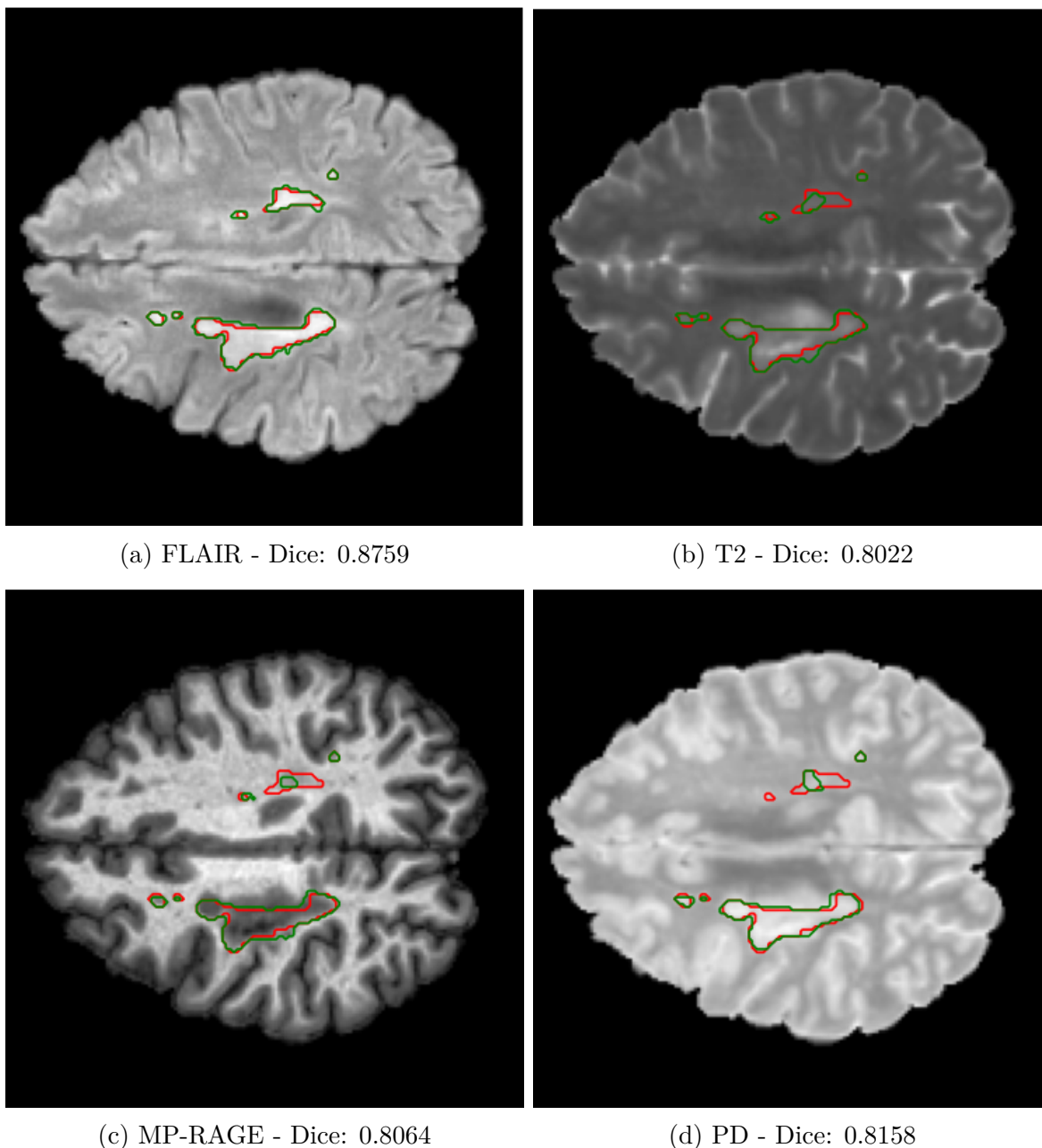


Figura 5.8: Exemplos de segmentação de lesões de esclerose múltipla com a VGG16-UNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

## 5.4 Experimento III: SegNet

O terceiro conjunto de experimentos deste capítulo consistiu na aplicação da arquitetura SegNet. Ela é um modelo simétrico, assim como a UNet, mas não possui *skip connections* nos blocos de convolução, como é mencionado na Subseção 4.3.3. As execuções realiza-

das aqui são semelhantes as anteriores, com o mesmo número de imagens e as mesmas configurações do modelo.

As Tabelas 5.5 e 5.6 exibem os resultados obtidos no conjunto de imagens de teste (279 imagens). É possível perceber que o aumento de dados melhorou o resultado do coeficiente Dice para todas as sequências. A especificidade também teve uma melhora nos resultados para FLAIR e PD com a técnica de aumento de dados, porém os valores da acurácia e da sensibilidade diminuíram para essas sequências. Dessa forma, pode-se afirmar que os experimentos com e sem essa técnica empataram.

As imagens T2 mostram um aumento no valor das métricas acurácia e sensibilidade com o método de aumento de dados. No entanto, esse comportamento não é observado para a sequência MP-RAGE, cujo resultado da acurácia diminuiu e o da sensibilidade aumentou com o uso do método. A especificidade se manteve constante para essas sequências em ambos os experimentos. Por meio dessa análise, nota-se que o experimento com aumento de dados obteve melhores resultados para as sequências T2 e MP-RAGE.

Tabela 5.5: Resultados obtidos para a arquitetura SegNet sem aumento de dados.

Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.5758</b>	<b>0.9971</b>	0.6806	<b>0.9989</b>
T2	0.5266	0.9965	<b>0.7111</b>	0.9982
MP-RAGE	0.5428	0.9970	0.6742	0.9988
PD	0.5181	0.9968	0.6397	0.9988

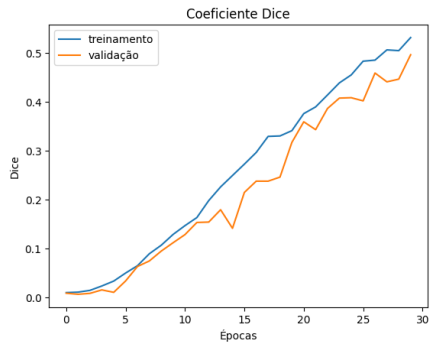
Tabela 5.6: Resultados obtidos para a arquitetura SegNet com aumento de dados.

Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	0.6024	<b>0.9970</b>	0.6648	0.9990
T2	<b>0.6129</b>	0.9968	<b>0.7554</b>	0.9982
MP-RAGE	0.5886	0.9969	0.6802	0.9988
PD	0.5380	0.9966	0.5466	<b>0.9992</b>

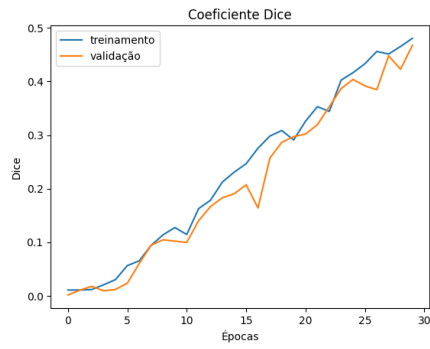
As Figuras 5.9 e 5.10 apresentam as curvas de coeficiente Dice dos experimentos sem e com o método de aumento de dados, respectivamente. Percebe-se que esse método introduziu um pouco de *overfitting* no modelo, comportamento evidenciado pelo afastamento gradual das curvas de treinamento de validação. Contudo, os dados da Tabela 5.6 mostram que isso não atrapalhou o treinamento e, na verdade, gerou resultados melhores para essa métrica.

As Figuras 5.11 e 5.12 mostram as máscaras de segmentação obtidas pela SegNet. O resultado individual dessas imagens para o coeficiente Dice superou significativamente as médias calculadas nas Tabelas 5.5 e 5.6 e o aumento de dados aumentou todos os valores dessa métrica. Entretanto, é notável que em todas as imagens algumas das lesões

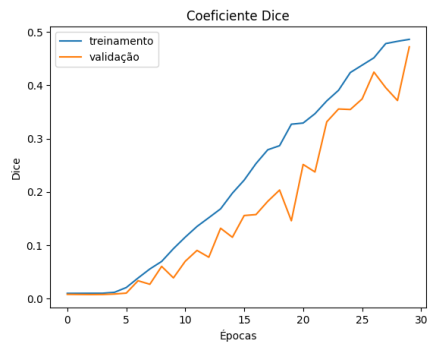




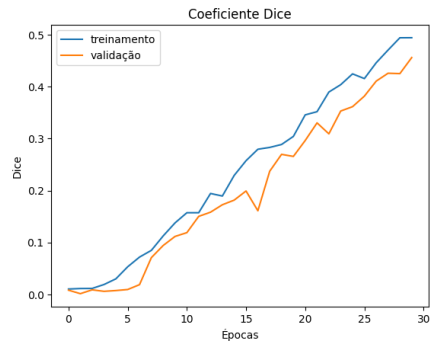
(a) FLAIR



(b) T2

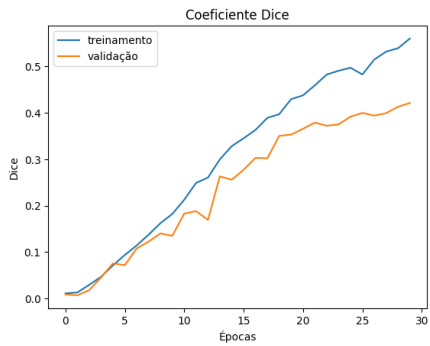


(c) MP-RAGE

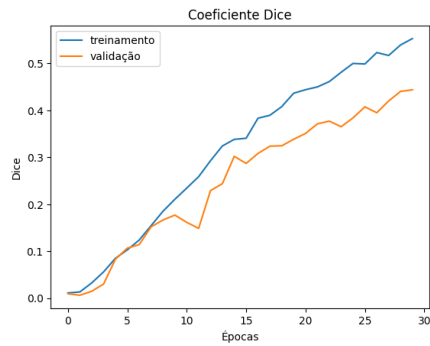


(d) PD

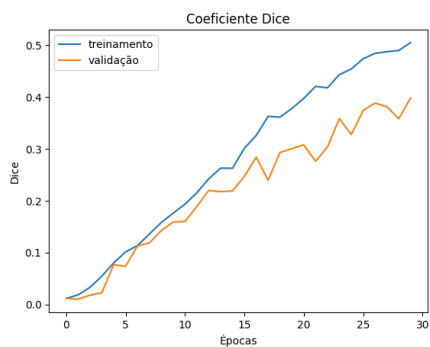
Figura 5.9: Curva de coeficiente Dice da SegNet sem aumento de dados.



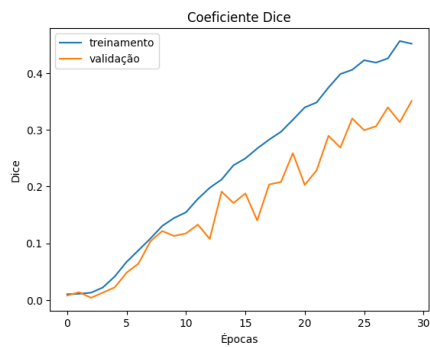
(a) FLAIR



(b) T2



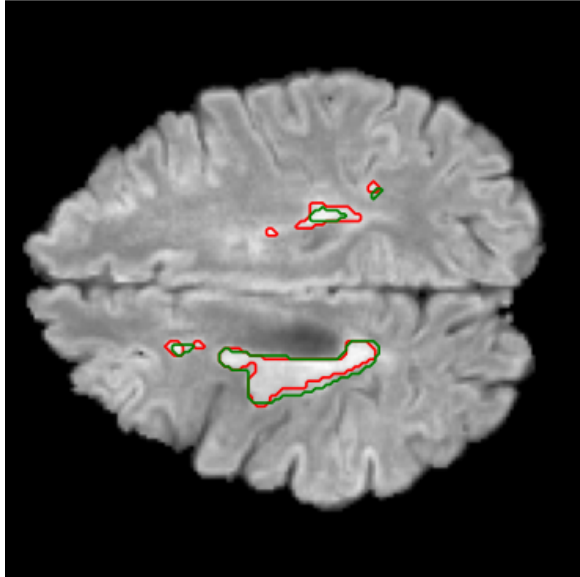
(c) MP-RAGE



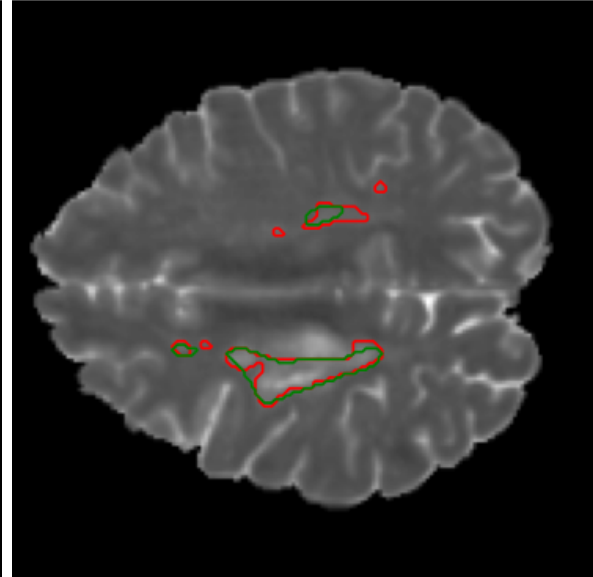
(d) PD

Figura 5.10: Curva de coeficiente Dice da SegNet com aumento de dados.

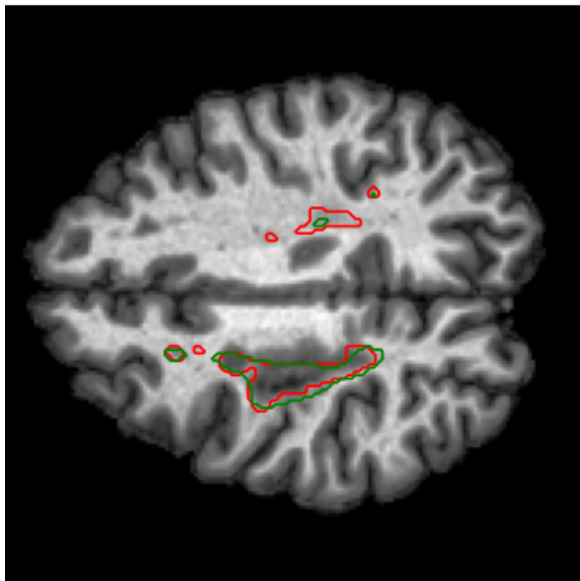
menores não foram corretamente delineadas e erroneamente classificadas como parte do plano de fundo (falsos negativos). Esse aspecto é ainda mais evidente após aplicar o aumento de dados nas sequências FLAIR e PD. Isso justifica os valores reduzidos obtidos na sensibilidade.



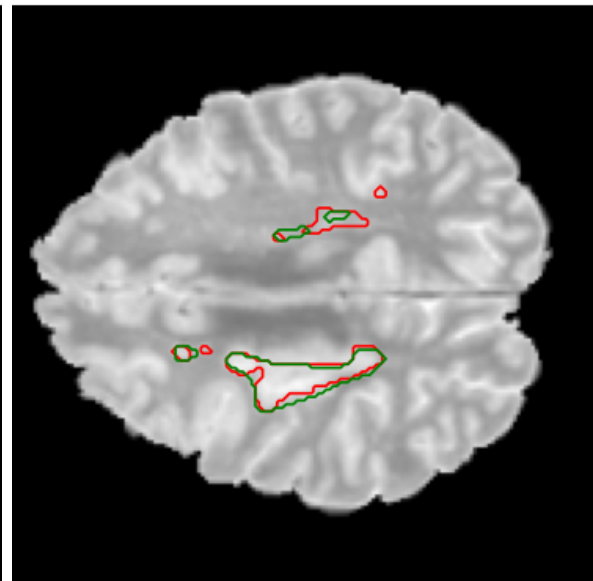
(a) FLAIR - Dice: 0.7197



(b) T2 - Dice: 0.6560



(c) MP-RAGE - Dice: 0.6583



(d) PD - Dice: 0.6421

Figura 5.11: Exemplos de segmentação de lesões de esclerose múltipla com a SegNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

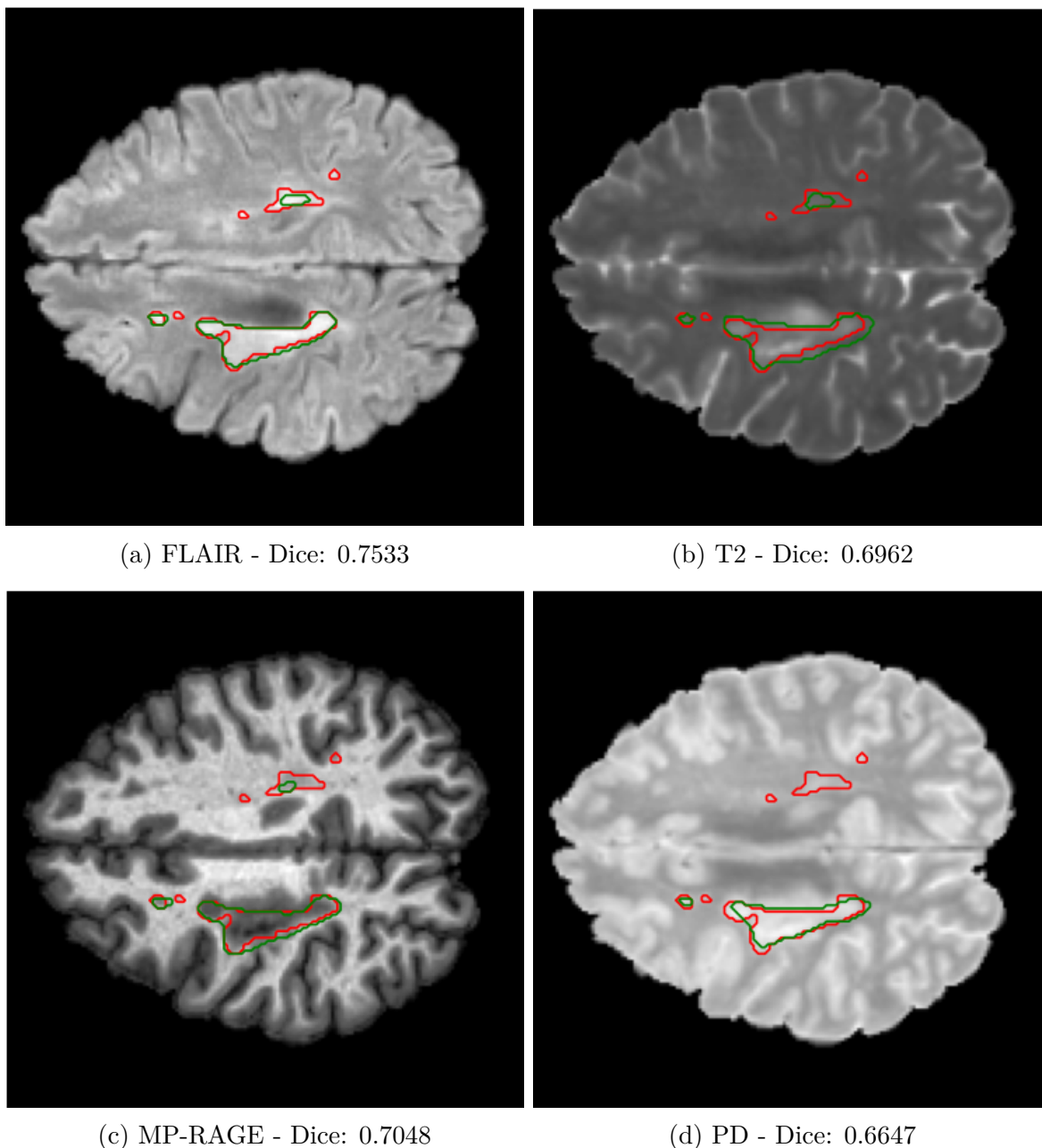


Figura 5.12: Exemplos de segmentação de lesões de esclerose múltipla com a SegNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

## 5.5 Experimento IV: ResUNet

O último conjunto de experimentos deste capítulo conta com a aplicação da arquitetura ResUNet. Essa CNN é uma variação da UNet e é formada por blocos residuais. Esses e outros detalhes do modelo são explicados na Subseção 2.5.4. Os experimentos realizados nesta seção utilizam os mesmos dados e configurações dos experimentos anteriores.

As Tabelas 5.7 e 5.8 detalham os resultados obtidos no conjunto de teste (279 imagens) para os experimentos sem e com a técnica de aumento de dados, respectivamente. Os dados mostram que essa técnica melhorou os resultados do coeficiente Dice e da sensibilidade para as sequências FLAIR e PD, enquanto a acurácia e a especificidade não sofreram alterações.

Para as imagens T2, o aumento de dados melhorou os resultados do coeficiente Dice, acurácia e especificidade e diminuiu apenas o da sensibilidade. A sequência MP-RAGE sofreu uma redução no resultado da especificidade com o método de aumento de dados. No entanto, os valores do coeficiente Dice e da sensibilidade aumentaram com o método e o resultado da acurácia se manteve constante. Dessa forma, o experimento com aumento de dados gerou melhores resultados para todas as sequências de ressonância magnética.

Tabela 5.7: Resultados obtidos para a arquitetura ResUNet sem aumento de dados.

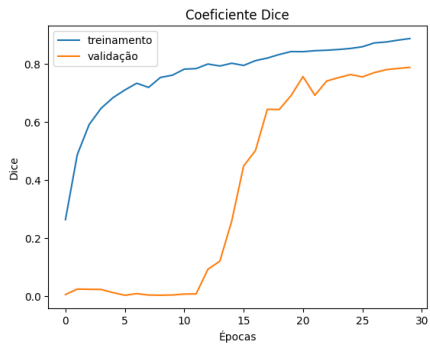
Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.8187</b>	<b>0.9984</b>	0.8208	<b>0.9994</b>
T2	0.7555	0.9977	<b>0.8498</b>	0.9986
MP-RAGE	0.7625	0.9979	0.7430	<b>0.9994</b>
PD	0.7795	0.9981	0.7798	0.9993

Tabela 5.8: Resultados obtidos para a arquitetura ResUNet com aumento de dados.

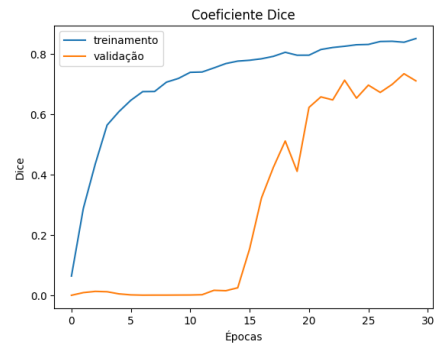
Sequências	Coeficiente Dice	Acurácia	Sensibilidade	Especificidade
FLAIR	<b>0.8268</b>	<b>0.9984</b>	0.8222	<b>0.9994</b>
T2	0.7814	0.9980	<b>0.8345</b>	0.9990
MP-RAGE	0.7793	0.9979	0.8311	0.9989
PD	0.7867	0.9981	0.7907	0.9993

As Figuras 5.13 e 5.14 mostram as curvas de coeficiente Dice dos experimentos sem e com o método de aumento de dados, respectivamente. É notável que o modelo sofre muito *overfitting* no início do treinamento, em todas as execuções, mas isso diminui à medida que as épocas aumentam e o aprendizado da rede se intensifica. O aumento de dados tem um forte impacto na redução inicial do *overfitting*, seguido por uma estabilização da curva de validação.

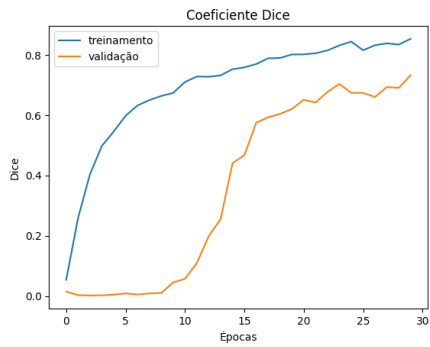
Nas Figuras 5.15 e 5.16 são apresentados alguns resultados da segmentação gerados pela ResUNet. É notável que o método de aumento de dados teve um impacto muito positivo e melhorou o resultado do coeficiente Dice para todas as imagens. No entanto, é visível que esse método prejudicou a segmentação de lesões pequenas na imagem T2, justificando a redução da sensibilidade.



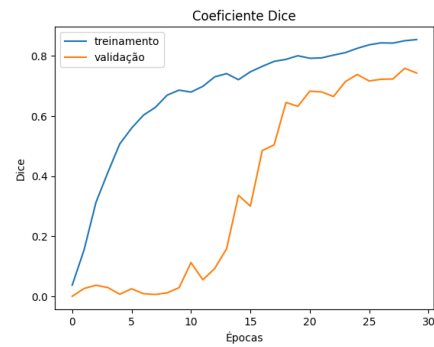
(a) FLAIR



(b) T2

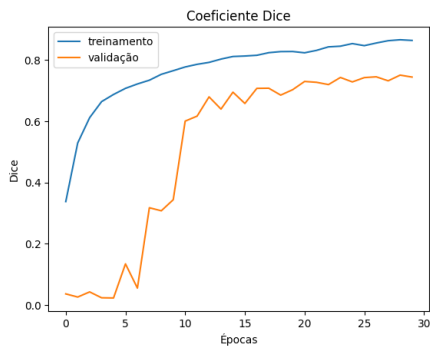


(c) MP-RAGE

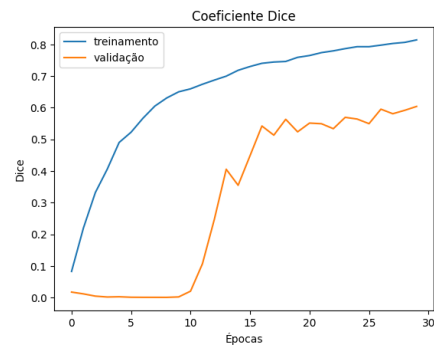


(d) PD

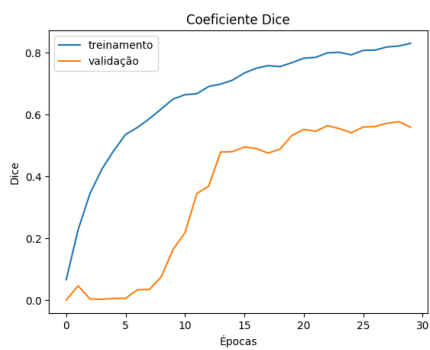
Figura 5.13: Curva de coeficiente Dice da ResUNet sem aumento de dados.



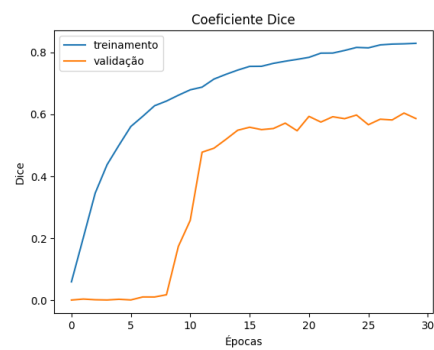
(a) FLAIR



(b) T2

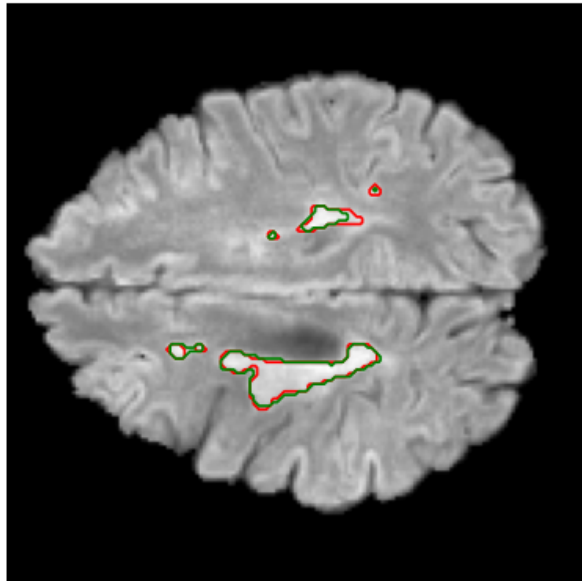


(c) MP-RAGE

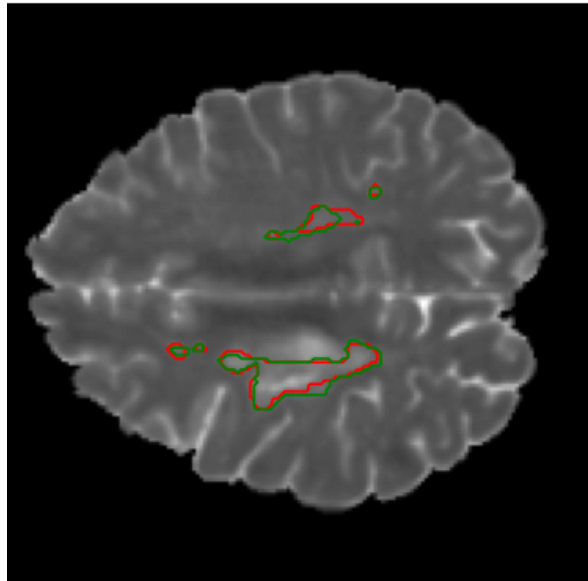


(d) PD

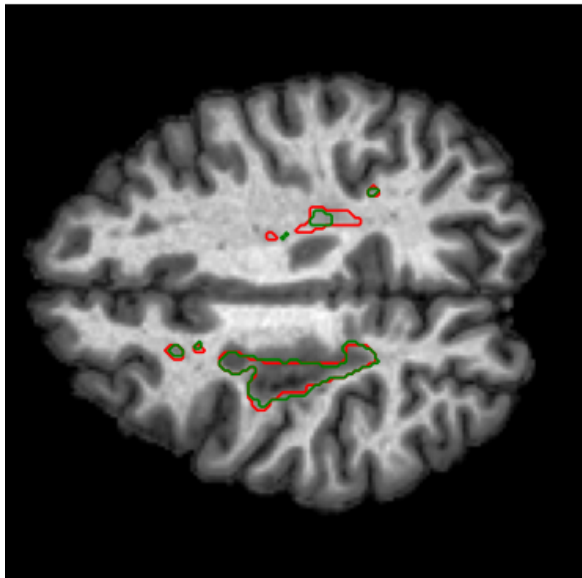
Figura 5.14: Curva de coeficiente Dice da ResUNet com aumento de dados.



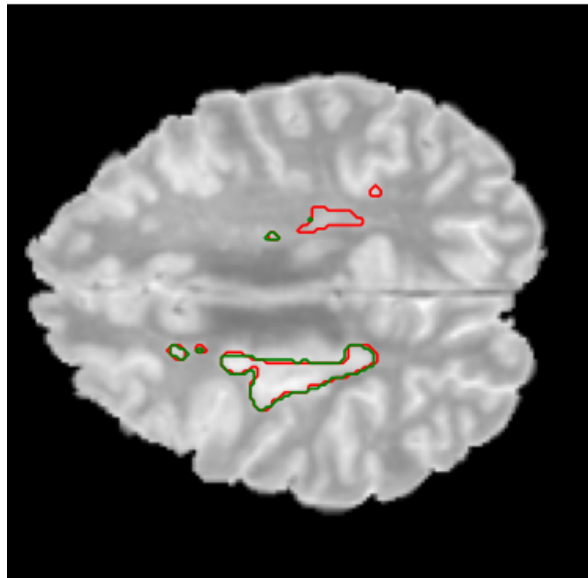
(a) FLAIR - Dice: 0.8823



(b) T2 - Dice: 0.7960



(c) MP-RAGE - Dice: 0.8115



(d) PD - Dice: 0.7994

Figura 5.15: Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet, sem o uso da técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

Na imagem MP-RAGE, observa-se que as regiões do plano de fundo erroneamente classificadas como lesões aumentaram com a aplicação do aumento de dados, fator que contribuiu para a redução da especificidade.

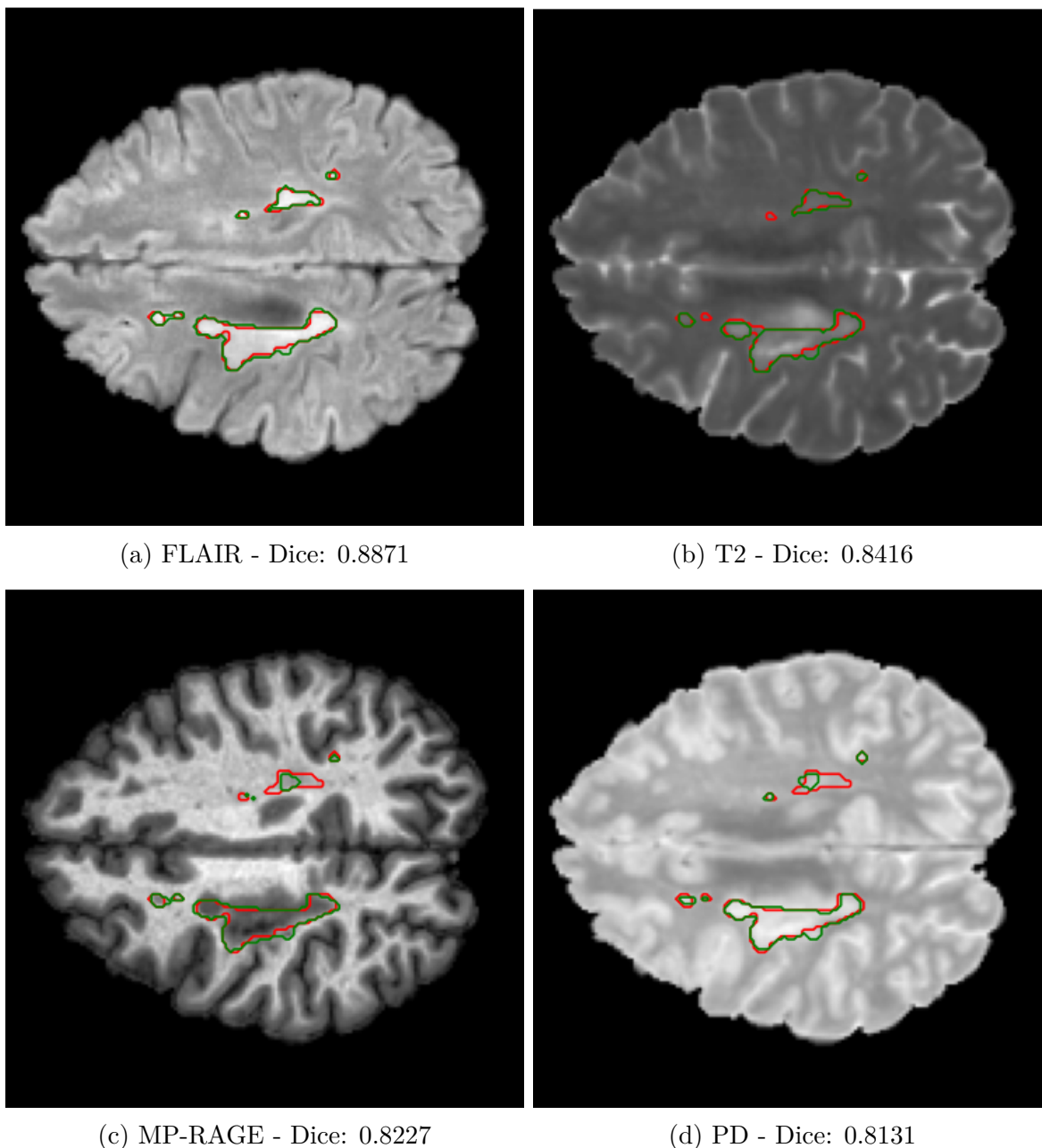


Figura 5.16: Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet e a técnica de aumento de dados. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

## 5.6 Síntese dos resultados

Nas seções anteriores foram apresentados os resultados alcançados durante os experimentos realizados neste trabalho, assim como uma análise dos métodos empregados e o seu impacto no desempenho das redes. Esta última seção exibe uma síntese das métricas obtidas para cada CNN e analisa quais modelos são mais eficientes na tarefa de segmentação

de lesões de esclerose múltipla. Para essa análise é usado o coeficiente Dice. Essa métrica indica a semelhança entre a imagem predita e o *ground-truth* com maior precisão e, por isso, ela foi escolhida como parâmetro para as comparações dos modelos.

A Tabela 5.9 exibe um resumo dos melhores resultados obtidos nos experimentos deste capítulo. Para cada sequência de ressonância magnética é destacado o maior valor calculado. É possível observar que a VGG16-UNet gerou ótimos resultados, comparada à UNet, com exceção das imagens PD. O *transfer learning* é um método muito eficaz na área de aprendizado de máquina e, de modo geral, afetou positivamente o desempenho da rede. O aumento de dados, no entanto, não melhorou o desempenho do modelo para todas as sequências, apenas para MP-RAGE e PD.

Ao comparar os resultados da SegNet com os dados da UNet e VGG16-UNet, é notável a diferença de valores. A SegNet tem um desempenho bem mais baixo, mostrando que esse modelo não é o mais adequado para a segmentação de lesões de esclerose múltipla. Uma possível explicação para esse comportamento é a ausência das *skip connections* na sua arquitetura, resultando em um aumento na perda de informações da imagem quando ela percorre múltiplas camadas.

Analisando os resultados das ResUNet em comparação à UNet, VGG16-UNet e SegNet, é perceptível que o desempenho desse modelo superou as outras arquiteturas. Esse comportamento se deve possivelmente as *skip connections*, cuja ausência prejudicou a eficiência da SegNet. A ResUNet, por outro lado, é favorecida pela presença de várias dessas operações na sua arquitetura. Elas estão presentes tanto nos blocos residuais quanto no caminho de expansão, como é citado na Subseção 4.3.4. Em virtude disso, os resultados melhoraram significativamente para quase todas as imagens. A única exceção é a sequência T2, cujo melhor resultado é para a VGG16-UNet, sem o método de aumento de dados. Com o uso desse método, a ResUNet obteve os melhores resultados para todos os conjuntos de imagens.

Tabela 5.9: Resultado do coeficiente Dice após aplicar as arquiteturas UNet, VGG16-UNet, Segnet e ResUNet para cada sequência de ressonância magnética.

Pré-processamento	Sequências	UNet	VGG16-UNet	SegNet	ResUNet
Sem aumento de dados	FLAIR	0.7984	0.8109	0.5758	<b>0.8187</b>
	T2	0.7450	<b>0.7607</b>	0.5266	0.7555
	MP-RAGE	0.7511	0.7600	0.5428	<b>0.7625</b>
	PD	0.7507	0.7438	0.5181	<b>0.7795</b>
Com aumento de dados	FLAIR	0.7932	0.8069	0.6024	<b>0.8268</b>
	T2	0.7536	0.7416	0.6129	<b>0.7814</b>
	MP-RAGE	0.7440	0.7706	0.5886	<b>0.7793</b>
	PD	0.7493	0.7726	0.5380	<b>0.7867</b>



# Capítulo 6

## Conclusão e trabalhos futuros

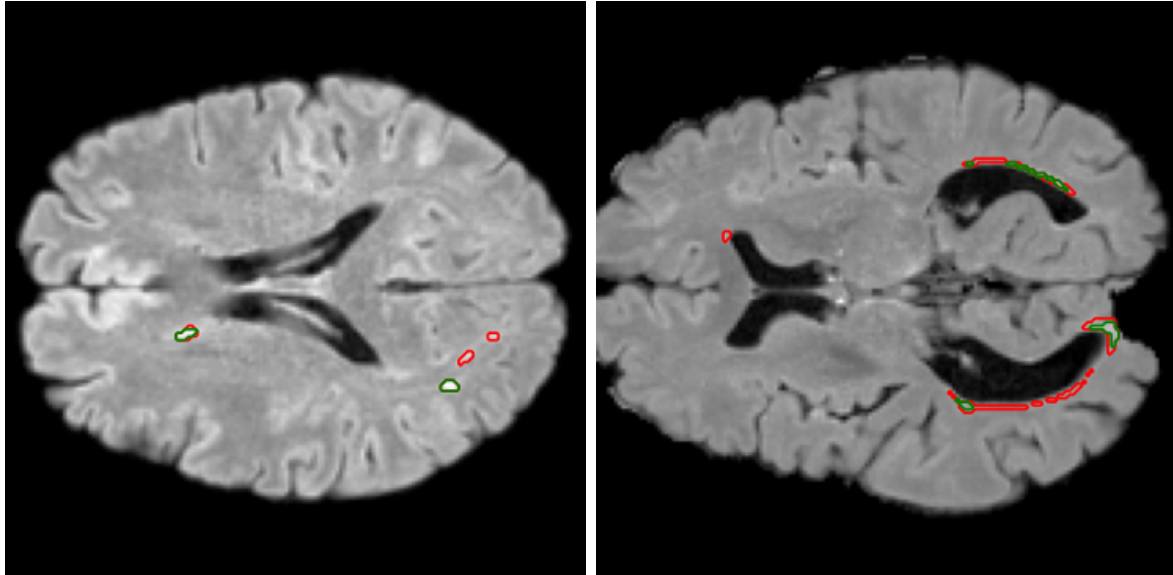
O objetivo deste trabalho foi auxiliar na segmentação de lesões de esclerose múltipla em imagens de ressonância magnética a partir da análise de técnicas de aprendizado de máquina e diferentes modelos de CNN. Para alcançar esse objetivo foram realizados experimentos com as técnicas de *transfer learning* e aumento de dados e com as arquiteturas UNet, VGG16-UNet, SegNet e ResUNet. Também foram utilizadas as sequências de ressonância magnética FLAIR, T2, MP-RAGE e PD.

Apesar do bom desempenho da UNet aliada ao *transfer learning*, representada pelo modelo VGG16-UNet, a ResUNet, com aplicação da técnica de aumento de dados durante a etapa de pré-processamento, obteve os melhores resultados e exigiu um tempo de treinamento menor, de aproximadamente 7h, em comparação as 12h de treinamento da VGG16-UNet. Ademais, o modelo gerou ótimos resultados para todas as sequências, alcançando um coeficiente Dice de 0.8268 para FLAIR, 0.7814 para T2, 0.7793 para MP-RAGE e 0.7867 para PD. Isso mostra que a ResUNet tem um bom desempenho aplicada em imagens de ressonância magnética com diferentes características. Os resultados também evidenciam que a sequência FLAIR tem mais impacto na segmentação de lesões e, por este motivo, é utilizada com mais frequência em estudos na área.

Dentre os problemas mais recorrentes nos estudos de segmentação de lesões de esclerose múltipla, predomina-se a limitação das base de dados, cujo número de imagens é insuficiente para treinar modelos com eficiência. O aumento de dados e o *transfer learning* foram aplicados a fim de minimizar essa limitação, e efetivamente aumentaram as métricas de desempenho. No entanto, não é possível garantir que as CNNs utilizadas obtenham resultados semelhantes ao serem expostas a um novo conjuntos de imagens.

Outra dificuldade encontrada em alguns experimentos foi a segmentação de lesões pequenas, erroneamente classificadas como falsos negativos, ou seja, como parte de outros tecidos do cérebro. Na Figura 6.1 são mostradas algumas imagens da sequência FLAIR cujo resultado da segmentação ficou abaixo do esperado. Em ambos os casos há lesões que

não foram identificadas pelo modelo. Isso acontece devido à desproporção entre o número de *pixels* que compõem o plano de fundo e o número de *pixels* que fazem parte de uma lesão, que é bem menor em relação à área total da imagem. Devido a esse desequilíbrio, as CNNs têm dificuldade em segmentar lesões pequenas.



(a) Dice: 0.6555

(b) Dice: 0.5369

Figura 6.1: Exemplos de segmentação de lesões de esclerose múltipla com a ResUNet e a técnica de aumento de dados para imagens FLAIR. É mostrada a sobreposição do contorno do *ground-truth*, em vermelho, e do contorno predito pelo modelo, em verde, na imagem original.

Para trabalhos futuros, o uso de bases de dados mais robustas, com disponibilidade de diferentes sequências de ressonância magnética e maior variedade de pacientes, permitiria validar se o desempenho dos modelos VGG16-UNet e ResUNet seria mantido. A variedade de sequências auxiliaria no desenvolvimento de uma rede neural com capacidade de boa generalização em qualquer tipo de imagem de ressonância magnética. Ademais, o uso de máscaras de segmentação fornecidas por outros profissionais da saúde também permitiria comparar o desempenho do modelo ao desempenho do método humano.

Por fim, a aplicação de diferentes transformações de aumento de dados, como injeção de ruído, recorte de imagem e alterações no brilho, pode auxiliar na redução de falsos negativos [42]. A alteração dos pesos das classes de uma CNN também pode auxiliar nesse processo [42].

# Referências

- [1] Azad, Reza, Nika Khosravi, Mohammad Dehghanmanshadi, Julien Cohen-Adad e Dorit Merhof: *Medical image segmentation on MRI images with missing modalities: A review*. arXiv preprint arXiv:2203.06217, 2022. ix, 7
- [2] *ISBI MS dataset*. <https://www.kaggle.com/datasets/marwa96/isbi-ms-dataset>. Acessado em 07/12/2023. ix, 26
- [3] McGinley, Marisa P, Carolyn H Goldschmidt e Alexander D Rae-Grant: *Diagnosis and treatment of multiple sclerosis: a review*. *Jama*, 325(8):765–779, 2021. 1, 2
- [4] *World Health Organization: Multiple sclerosis*. <https://www.who.int/news-room/fact-sheets/detail/multiple-sclerosis>. Acessado em 23/11/2023. 1
- [5] Bichueti, Denis Bernardi: *A vida com esclerose múltipla é um desafio a superar*. <https://sp.unifesp.br/epe/desc/noticias/esclerose-multippla-2021>. Acessado em 06/12/2023. 1
- [6] Goldenberg, Marvin M: *Multiple sclerosis review*. *Pharmacy and therapeutics*, 37(3):175, 2012. 2
- [7] Zeng, Chenyi, Lin Gu, Zhenzhong Liu e Shen Zhao: *Review of deep learning approaches for the segmentation of multiple sclerosis lesions on brain mri*. *Frontiers in Neuroinformatics*, 14:610967, 2020. 2
- [8] Shoeibi, Afshin, Marjane Khodatars, Mahboobeh Jafari, Parisa Moridian, Mitra Rezaei, Roohallah Alizadehsani, Fahime Khozeimeh, Juan Manuel Gorris, Jónathan Heras, Maryam Panahiazar *et al.*: *Applications of deep learning techniques for automated multiple sclerosis detection using magnetic resonance imaging: A review*. *Computers in Biology and Medicine*, 136:104697, 2021. 2
- [9] La Rosa, Francesco, Ahmed Abdulkadir, Mário João Fartaria, Reza Rahmanzadeh, Po Jui Lu, Riccardo Galbusera, Muhamed Barakovic, Jean Philippe Thiran, Cristina Granziera e Merixtell Bach Cuadra: *Multiple sclerosis cortical and wm lesion segmentation at 3t mri: a deep learning method based on flair and mp2rage*. *NeuroImage: Clinical*, 27:102335, 2020. 2, 3, 19, 23
- [10] Ghosh, Anirudha, Abu Sufian, Farhana Sultana, Amlan Chakrabarti e Debashis De: *Fundamental concepts of convolutional neural network*. *Recent trends and advances in artificial intelligence and Internet of Things*, páginas 519–567, 2020. 2, 8, 9, 10, 11, 12, 17, 28

- [11] Minaee, Shervin, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz e Demetri Terzopoulos: *Image segmentation using deep learning: A survey*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(7):3523–3542, 2021. 2, 7
- [12] Goodfellow, Ian, Yoshua Bengio e Aaron Courville: *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>. 3, 7, 8, 9, 10, 15
- [13] Cheng, Dorothy e Edmund Y Lam: *Transfer learning U-net deep learning for lung ultrasound segmentation*. arXiv preprint arXiv:2110.02196, 2021. 3, 14, 21, 23, 28
- [14] Hashemi, Maryam, Mahsa Akhbari e Christian Jutten: *Delve into multiple sclerosis (ms) lesion exploration: A modified attention u-net for ms lesion segmentation in brain MRI*. Computers in Biology and Medicine, 145:105402, 2022, ISSN 0010-4825. <https://www.sciencedirect.com/science/article/pii/S0010482522001949>. 3, 20, 23
- [15] David C Preston, MD: *Magnetic Resonance Imaging (MRI) of the brain and spine: Basics*. <https://case.edu/med/neurology/NR/MRI%20Basics.htm>. Acessado em 12/10/2023. 4, 5, 6
- [16] Berger, Abi: *How does it work?: Magnetic resonance imaging*. BMJ: British Medical Journal, 324(7328):35, 2002. 4
- [17] Health, UF: *About MRI*. <https://merritt.biochem.med.ufl.edu/about-mri/>. Acessado em 20/10/2023. 5
- [18] Filippi, Massimo, Paolo Preziosa, Brenda L Banwell, Frederik Barkhof, Olga Ciccarelli, Nicola De Stefano, Jeroen JG Geurts, Friedemann Paul, Daniel S Reich, Ahmed T Toosy *et al.*: *Assessment of lesions on magnetic resonance imaging in multiple sclerosis: practical guidelines*. Brain, 142(7):1858–1875, 2019. 6
- [19] Wang, Jinghua, Lili He, Hairong Zheng e Zhong Lin Lu: *Optimizing the magnetization-prepared rapid gradient-echo (MP-RAGE) sequence*. PloS one, 9(5):e96899, 2014. 6
- [20] Li, Xiangrui, Paul S. Morgan, John Ashburner, Jolinda Smith e Christopher Rorden: *The first step for neuroimaging data analysis: Dicom to nifti conversion*. Journal of Neuroscience Methods, 264:47–56, 2016, ISSN 0165-0270. <https://www.sciencedirect.com/science/article/pii/S0165027016300073>. 6
- [21] Artacho, Bruno e Andreas Savakis: *Waterfall atrous spatial pooling architecture for efficient semantic segmentation*. Sensors, 19(24):5361, 2019. 8
- [22] Ferreira, Mateus F de C, Paula D Portella, Juliana F de Souza, Bruna C Dias, Luciana R da S Assunção e Lucas F de Oliveira: *Avaliação do uso redes neurais convolucionais para identificação de lesões cáries dentárias*. Em *Anais do XXIII Simpósio Brasileiro de Computação Aplicada à Saúde*, páginas 473–478. SBC, 2023. 9

- [23] Code, Papers with: *Max pooling*. <https://paperswithcode.com/method/max-pooling>. Acessado em 17/10/2023. 11
- [24] Ronneberger, Olaf, Philipp Fischer e Thomas Brox: *U-net: Convolutional networks for biomedical image segmentation*. Em *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, páginas 234–241. Springer, 2015. 13, 28, 30
- [25] Shi, Jiyuan, Ji Dang, Mida Cui, Rongzhi Zuo, Kazuhiro Shimizu, Akira Tsunoda e Yasuhiro Suzuki: *Improvement of damage segmentation based on pixel-level data balance using vgg-unet*. *Applied Sciences*, 11(2):518, 2021. 14
- [26] Badrinarayanan, Vijay, Alex Kendall e Roberto Cipolla: *Segnet: A deep convolutional encoder-decoder architecture for image segmentation*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. 14, 28
- [27] Zhang, Zhengxin, Qingjie Liu e Yunhong Wang: *Road extraction by deep residual u-net*. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018. 15, 16, 28
- [28] Valverde, Sergi, Mariano Cabezas, Eloy Roura, Sandra González-Villà, Deborah Pareto, Joan C. Vilanova, Lluís Ramió-Torrentà, Àlex Rovira, Arnau Oliver e Xavier Lladó: *Improving automated multiple sclerosis lesion segmentation with a cascaded 3d convolutional neural network approach*. *NeuroImage*, 155:159–168, 2017, ISSN 1053-8119. <https://www.sciencedirect.com/science/article/pii/S1053811917303270>. 19, 23
- [29] Afzal, HM, Suhuai Luo, Saadallah Ramadan, Jeannette Lechner-Scott, Mohammad Ruhul Amin, Jiaming Li e M Kamran Afzal: *Automatic and robust segmentation of multiple sclerosis lesions with convolutional neural networks*. *Computers, Materials & Continua*, 66(1), 2021. 19, 23
- [30] Gessert, Nils, Julia Krüger, Roland Opfer, Ann Christin Ostwaldt, Praveena Manogaran, Hagen H Kitzler, Sven Schippling e Alexander Schlaefer: *Multiple sclerosis lesion activity segmentation with attention-guided two-path cnns*. *Computerized Medical Imaging and Graphics*, 84:101772, 2020. 20, 23
- [31] Acar, Züleyha Yılmaz, Fatih Başçiftçi e Ahmet Hakan Ekmekci: *A convolutional neural network model for identifying multiple sclerosis on brain FLAIR MRI*. *Sustainable Computing: Informatics and Systems*, 35:100706, 2022. 20, 23
- [32] Kamraoui, Reda Abdellah, Vinh Thong Ta, Thomas Tourdias, Boris Mansencal, José V Manjon e Pierrick Coup: *Deeplesionbrain: Towards a broader deep-learning generalization for multiple sclerosis lesion segmentation*. *Medical Image Analysis*, 76:102312, 2022. 20, 23
- [33] Gaj, Sibaji, Daniel Ontaneda e Kunio Nakamura: *Automatic segmentation of gadolinium-enhancing lesions in multiple sclerosis using deep learning from clinical mri*. *PloS one*, 16(9):e0255939, 2021. 21, 23

- [34] Essa, Ehab, Doaa Aldesouky, Sherif E Hussein e Magdi Zakria Rashad: *Neuro-fuzzy patch-wise R-CNN for multiple sclerosis segmentation*. *Medical & Biological Engineering & Computing*, 58:2161–2175, 2020. 21, 23
- [35] Rahman, Hameedur, Tanvir Fatima Naik Bukht, Azhar Imran, Junaid Tariq, Shanshan Tu e Abdulkareem Alzahrani: *A deep learning approach for liver and tumor segmentation in ct images using resunet*. *Bioengineering*, 9(8):368, 2022. 21, 23
- [36] Gheshlaghi, Saba Heidari, Amin Ranjbar, Amir Abolfazl Suratgar, Mohammad Bagher Menhaj e Fardin Faraji: *A superpixel segmentation based technique for multiple sclerosis lesion detection*. *arXiv preprint arXiv:1907.03109*, 2019. 22, 23
- [37] Cetin, Ozdemir, Volkan Seymen e Unal Sakoglu: *Multiple sclerosis lesion detection in multimodal mri using simple clustering-based segmentation and classification*. *Informatics in Medicine Unlocked*, 20:100409, 2020. 22, 23
- [38] Carass, Aaron, Snehashis Roy, Amod Jog, Jennifer L Cuzzocreo, Elizabeth Magrath, Adrian Gherman, Julia Button, James Nguyen, Ferran Prados, Carole H Sudre *et al.*: *Longitudinal multiple sclerosis lesion segmentation: resource and challenge*. *NeuroImage*, 148:77–102, 2017. 24
- [39] Tustison, Nicholas J, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich e James C Gee: *N4ITK: improved N3 bias correction*. *IEEE transactions on medical imaging*, 29(6):1310–1320, 2010. 25
- [40] Carass, Aaron, M Bryan Wheeler, Jennifer Cuzzocreo, Pierre Louis Bazin, Susan S Bassett e Jerry L Prince: *A joint registration and segmentation approach to skull stripping*. Em *2007 4th IEEE international symposium on biomedical imaging: from nano to macro*, páginas 656–659. IEEE, 2007. 25
- [41] Shiee, Navid, Pierre Louis Bazin, Jennifer L Cuzzocreo, Chuyang Ye, Bhaskar Kishore, Aaron Carass, Peter A Calabresi, Daniel S Reich, Jerry L Prince e Dzung L Pham: *Reconstruction of the human cerebral cortex robust to white matter lesions: method and validation*. *Human brain mapping*, 35(7):3385–3401, 2014. 25
- [42] Akkus, Zeynettin, Alfiya Galimzianova, Assaf Hoogi, Daniel L Rubin e Bradley J Erickson: *Deep learning for brain MRI segmentation: state of the art and future directions*. *Journal of Digital Imaging*, 30:449–459, 2017. 52