



UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA
DEPARTAMENTO DE MATEMÁTICA APLICADA



NATHAN OLIVEIRA CARVALHO DA SILVA

Aplicação de Redes Neurais Convolucionais na Álgebra de Clifford na Classificação de Imagens Dermatoscópicas

Campinas
28/06/2023

NATHAN OLIVEIRA CARVALHO DA SILVA

Aplicação de Redes Neurais Convolucionais na Álgebra de Clifford na Classificação de Imagens Dermatoscópicas

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do(a) Prof. Marcos Eduardo Ribeiro do Valle Mesquita.

Resumo

O projeto visa explorar o uso de redes neurais convolucionais hipercomplexas na classificação de lesões de pele. Utilizando técnicas da Álgebra de Clifford, busca-se melhorar a precisão e a capacidade de detecção de diferentes tipos de manchas, como melanomas, nevos normais e anormais. O objetivo é aprimorar os sistemas de diagnóstico auxiliados por computador e contribuir para um tratamento mais eficaz.

Sumário

| | | |
|----------|---|-----------|
| 1 | Introdução | 5 |
| 1.1 | Problema de câncer de pele | 5 |
| 2 | Álgebra de Clifford | 6 |
| 3 | Redes Neurais | 7 |
| 3.1 | Camada Densa | 8 |
| 3.2 | Camada de Convolução | 9 |
| 3.2.1 | Filtro | 9 |
| 3.2.2 | Real | 10 |
| 3.2.3 | Clifford | 11 |
| 3.3 | Camada de Pooling | 12 |
| 4 | Metodologia | 13 |
| 4.1 | Descrição do Conjunto de Dados | 13 |
| 4.2 | Pré-processamento | 15 |
| 4.3 | Representação dos Dados | 16 |
| 4.4 | Arquitetura das Redes | 17 |
| 4.5 | Divisão do Conjunto de Dados | 18 |
| 4.6 | Configurações de Treinamento | 19 |
| 4.7 | Métricas de Desempenho | 19 |
| 5 | Resultados e Discussão | 20 |
| 5.1 | Comparativo do Tamanho das Redes | 21 |
| 5.2 | Discussão sobre o desempenho e viés de cada tipo de rede | 21 |
| 5.3 | Análise do número de parâmetros e custo computacional | 23 |
| 6 | Conclusão | 23 |
| 6.1 | Resumo dos principais resultados e descobertas do estudo | 23 |
| 6.2 | Limitações do estudo e possíveis direções futuras de pesquisa | 24 |

1 Introdução

O câncer de pele é uma preocupação global de saúde, com alta incidência e impacto significativo na qualidade de vida dos pacientes. A detecção precoce e a classificação precisa das lesões de pele são fundamentais para um tratamento eficaz e melhores resultados clínicos. No entanto, a classificação precisa dessas lesões é um desafio para os dermatologistas, devido à complexidade e variabilidade das características das manchas de pele.

Uma abordagem promissora para auxiliar na classificação de imagens de manchas de pele é o uso de redes neurais convolucionais (CNNs) [2] [5]. As CNNs são uma forma especializada de redes neurais que podem aprender e extrair características relevantes das imagens. Elas têm sido amplamente utilizadas na área de visão computacional para classificação de imagens e têm apresentado resultados promissores em várias aplicações, incluindo a detecção de doenças.

Além disso, a utilização da Álgebra de Clifford tem ganhado destaque na área de processamento de informações. A Álgebra de Clifford é uma teoria matemática que permite o processamento de informações complexas, baseada em objetos multivetoriais. Essa abordagem pode fornecer uma representação mais rica e abrangente das informações contidas nas imagens, incluindo aspectos geométricos e estruturais.

Neste contexto, este projeto de pesquisa busca explorar o potencial das redes neurais convolucionais aliadas à Álgebra de Clifford, que têm mostrado resultados promissores em problemas de classificação de imagens médicas, como evidenciado em estudos anteriores [6]. Nosso objetivo é desenvolver e comparar modelos de CNNs convencionais e CNNs com valores hipercomplexos, utilizando a Álgebra de Clifford. Pretendemos analisar o desempenho desses modelos em termos de acurácia, precisão e sensibilidade na classificação de manchas normais, anormais e melanomas.

Ao final deste estudo, esperamos contribuir para aprimorar as técnicas de classificação de manchas de pele, facilitando a detecção precoce de cânceres de pele potencialmente perigosos e auxiliando os profissionais de saúde na tomada de decisões clínicas.

1.1 Problema de câncer de pele

O câncer de pele é um dos tipos mais comuns de câncer em todo o mundo, e a detecção precoce desempenha um papel crucial na eficácia do tratamento e na sobrevivência dos pacientes.

A classificação precisa das lesões de pele é essencial para identificar casos normais, anormais e potencialmente malignos, como melanomas. No entanto, essa tarefa pode ser desafiadora para dermatologistas, pois requer experiência e conhecimento aprofundado.

A abordagem convencional para a classificação de lesões de pele envolve a avaliação visual por dermatologistas, o que pode levar a variações de diagnóstico e atrasos no tratamento. Com o avanço da tecnologia e o aumento da disponibilidade de dados de imagens de lesões de pele, surge uma oportunidade para aplicar técnicas de aprendizado de máquina e visão computacional para auxiliar na classificação automática e precisa.

2 Álgebra de Clifford

Uma Álgebra de Clifford é definida a partir de um espaço vetorial de dimensão finita baseado em números reais ($V = \mathbb{R}^n$). O objetivo é tornar o espaço vetorial mais completo, adicionando escalares e multivetores. Os multivetores ampliam a ideia de vetores, pois possuem propriedades geométricas como área e volume, além de serem formados a partir da multiplicação de vetores [4].

Seja $\{e_1, e_2, \dots, e_n\}$ uma base ortonormal de \mathbb{R}^n . É possível definir a álgebra de Clifford $Cl(p, q)$, onde $p + q = n$, a partir da definição da multiplicação como uma operação associativa que satisfaz as seguintes condições:

- $e_i^2 = \begin{cases} +1, & \text{se } i = 1, \dots, p \\ -1, & \text{se } i = p + 1, \dots, n \end{cases}$
- $e_i \cdot e_j = -e_j \cdot e_i$, se $i \neq j$

Com essas propriedades é possível construir todos os produtos $e_{i_1 i_2 \dots i_r} = e_{i_1} \cdot e_{i_2} \dots e_{i_r}$, onde $\{i_1, i_2, \dots, i_r\} \subset \{1, 2, \dots, n\}$. Por exemplo, para $Cl(0, 2)$, $e_{121} = e_1 \cdot e_2 \cdot e_1 = (e_1 \cdot e_2) \cdot e_1 = (-e_2 \cdot e_1) \cdot e_1 = -e_2 \cdot (e_1 \cdot e_1) = -e_2 \cdot (-1) = e_2$

A Álgebra de Clifford $Cl(p, q)$ pode então ser definida como o espaço vetorial gerado pelos produtos de multivetores e_A , onde A é cada um dos subconjuntos de $\{1, 2, \dots, n\}$ escritos em ordem crescente, incluindo o conjunto vazio, que é considerado como o elemento escalar unitário, $e_\emptyset = 1$. Dessa forma, $Cl(p, q)$ possui dimensão 2^n . E todos os vetores de $Cl(p, q)$ pertencem ao espaço gerado por:

$$\left\{ \sum_{\lambda \in \Lambda} \alpha_{\lambda} e_{\lambda} : \alpha_{\lambda} \in \mathbb{R}, \lambda \in \Lambda \right\}, \quad (1)$$

onde Λ é o conjunto de todos os subconjuntos ordenados de $\{1, 2, \dots, n\}$.

Logo, considerando $n = 2$ e $\{e_1, e_2\}$ sendo uma base de vetores ortonormais em \mathbb{R}^2 , o conjunto $I = \{1, e_1, e_2, e_{12}\}$ é uma base para a Álgebra de Clifford $Cl(p, q)$, onde 1 é o elemento escalar, e_1 e e_2 são vetores básicos e e_{12} é um multivetor.

Por exemplo, é possível escrever a tabela de produto da álgebra $Cl(1, 1)$, onde se tem $e_1^2 = 1$ e $e_2^2 = -1$:

| | | | | |
|------------|-----------|----------|----------|-----|
| $Cl(1, 1)$ | e_1 | e_2 | e_{12} | (2) |
| e_1 | 1 | e_{12} | e_2 | |
| e_2 | $-e_{12}$ | -1 | e_1 | |
| e_{12} | $-e_2$ | $-e_1$ | 1 | |

Aqui nesse estudo, serão utilizadas as Álgebras de Clifford de 4 dimensões como $Cl(2, 0)$, $Cl(1, 1)$ e $Cl(0, 2)$, pois essas se mostraram as mais adequadas para a aplicação em processamento de imagens.

Assim, para as outras álgebras tem se as seguintes tabelas:

| | | | | | | | |
|------------|-----------|----------|----------|------------|-----------|----------|----------|
| $Cl(2, 0)$ | e_1 | e_2 | e_{12} | $Cl(0, 2)$ | e_1 | e_2 | e_{12} |
| e_1 | 1 | e_{12} | e_2 | e_1 | -1 | e_{12} | $-e_2$ |
| e_2 | $-e_{12}$ | 1 | $-e_1$ | e_2 | $-e_{12}$ | -1 | e_1 |
| e_{12} | $-e_2$ | e_1 | -1 | e_{12} | e_2 | $-e_1$ | -1 |

A Álgebra $Cl(0, 2)$ é isomórfica em relação aos quaternions enquanto as álgebras $Cl(2, 0)$ e $Cl(1, 1)$ são isomórficas aos coquaternions.

3 Redes Neurais

As redes neurais artificiais (Artificial Neural Networks, ANNs) são modelos matemáticos inspirados na estrutura do sistema nervoso biológico. Eles consistem em vários neurônios interconectados que trabalham juntos para resolver tarefas complexas, como classificação de imagens ou processamento de linguagem natural.

Cada neurônio na rede recebe entradas, aplica uma função de ativação e produz uma saída. As entradas podem ser valores de pixels de uma imagem, palavras em uma frase ou qualquer outra informação relevante para a tarefa em questão. As funções de ativação são usadas para modelar o comportamento do neurônio, geralmente sigmóide ou linear retificada.

Neste estudo, seguimos o paradigma de aprendizado supervisionado, no qual os dados são compostos por atributos e uma variável "alvo". No contexto deste trabalho, os atributos são os pixels das imagens, que atuam como valores de entrada, e a variável alvo representa a classe da mancha presente na imagem. O objetivo do treinamento da rede é ajustar seus parâmetros de modo que a saída se aproxime ao máximo da variável alvo.

3.1 Camada Densa

Uma camada densa é uma parte da estrutura de uma rede neural artificial que tem como objetivo processar os dados de entrada e gerar uma representação mais elaborada e abstrata dos mesmos, através de combinações lineares e não-lineares

A entrada para uma camada densa é um vetor de características, que é produzido pela camada anterior. Este vetor é multiplicado por uma matriz de pesos, que é aprendida durante o processo de treinamento da rede neural. Além disso, há um termo de bias adicionado à equação.

A saída da camada densa é produzida por uma função de ativação, aplicada sobre o resultado da multiplicação do vetor de características com a matriz de pesos e adição do termo de bias, que é usada para introduzir não-linearidade na representação produzida. Algumas das funções de ativação mais comuns incluem a função sigmoide, a tangente hiperbólica e a ReLU (função linear retificada). Neste caso específico, será utilizada a ReLU, cuja forma é representada por:

$$g(x) = \max(0, x) \quad (3)$$

O cálculo matemático para um neurônio de uma rede com números reais pode ser expressado como segue:

$$y_i = g \left(b_i + \sum_{j=1}^N w_{ij}x_j \right), \quad (4)$$

onde x_j é o vetor das características, w_{ij} matriz dos pesos, b_i é o bias, $g(\cdot)$ a função de ativação e y_i é a saída do neurônio j .

A camada com Álgebra de Clifford é análoga. Tendo em vista que os pesos, bias e inputs são representados por números hipercomplexos, então a operação da multiplicação do vetor de características com a matriz de pesos e adição do termo de bias também será um número hipercomplexo. A aplicação da função de ativação pode ser representado matematicamente da seguinte forma:

$$g\left(\sum_{\lambda} \alpha_{\lambda} e_{\lambda}\right) = \sum_{\lambda} g_{\mathbb{R}}(\alpha_{\lambda}) \gamma_{\lambda}, \quad (5)$$

ou seja, a aplicação da função em um número hipercomplexo é simplesmente a aplicação da função real em cada coeficiente da combinação linear.

3.2 Camada de Convolução

A camada convolucional é a peça-chave de uma CNN. Ela é uma forma específica de camada com parâmetros treináveis organizados em estruturas espaciais conhecidas como filtros. Estes filtros permitem que a rede processe dados localmente, aprendendo padrões locais. Os neurônios na primeira camada convolucional estão conectados apenas aos pixels em seus campos receptivos e cada neurônio em camadas subsequentes apenas a um pequeno retângulo na camada anterior. Isso permite que a rede concentre-se em recursos de baixo nível (como retas e curvas) e os transforme em recursos de maior nível. As CNNs são eficazes em reconhecimento de imagem devido à sua natureza espacial de aprendizado [1].

3.2.1 Filtro

Um filtro de convolução é uma matriz de pesos que “desliza” sobre a imagem de entrada com o objetivo de identificar características relevantes. A operação de multiplicação dos pesos do filtro com os valores dos pixels da imagem é conhecida como "convolução". Além disso, existem alguns hiperparâmetros que precisam ser definidos para o treinamento do modelo, tais como:

- Tamanho do filtro: que representa o número de pixels que são cobertos em cada aplicação do filtro. Por exemplo, um filtro de tamanho (3, 3) cobre 9 pixels. O tamanho do

filtro pode afetar a capacidade de capturar características complexas, mas também pode aumentar o custo computacional.

- **Padding:** número de pixels adicionados ao redor da imagem de entrada antes da aplicação do filtro, que visa preservar informações na borda.
- **Stride:** número de pixels deslocados entre cada aplicação do filtro. Quanto menor for o stride, maior será o número de operações de convolução, o que também aumentará a demanda computacional.
- **Número de filtros:** quantidade de filtros de convolução utilizados em uma camada da rede neural, permitindo capturar mais informações.

3.2.2 Real

Matematicamente, para uma rede com valores reais, é possível escrever o processo de convolução como:

$$(I * F)(p, k) = \sum_{c=1}^C \sum_{q \in D} I(p + S(q), c) F(q, c, k), \quad (6)$$

onde I é a imagem, que possui C canais de características/cores. A intensidade do c -ésimo canal da imagem I no pixel p é representada por $I(p, c) \in \mathbb{R}$.

Os filtros utilizados nessa operação também possuem C canais, correspondendo ao número de canais da imagem I . O domínio espacial dos filtros é representado por D .

Os pesos de uma camada de convolução com K filtros reais são organizados em um array F , onde $F(q, c, k)$ representa o valor do c -ésimo canal e do k -ésimo filtro no ponto $q \in D$. Aqui, c varia de 1 a C , representando os diferentes canais da imagem, e k varia de 1 a K , representando os diferentes filtros.

Além disso, a translação vertical e horizontal do filtro é representada por $S(q)$. Essa translação é análoga ao conceito de “janela deslizante” usado na convolução.

Portanto, a intensidade do k -ésimo canal de característica da saída de uma camada de convolução em um pixel p é definida por:

$$J(p, k) = g(b(k) + (I * F)(p, k)) \quad (7)$$

3.2.3 Clifford

Para a Álgebra de Clifford deve haver uma adaptação, pois os os inputs, pesos dos filtros e bias são hipercomplexos. Começando com:

$$(I^{(h)} * F^{(h)})(p, k) = \sum_{c=1}^C \sum_{q \in D} I^{(h)}(p + S(q), c) F^{(h)}(q, c, k), \quad (8)$$

em que o sobrescrito apenas indica que são uma imagem e um filtro hipercomplexo.

Digamos que a imagem e o filtro podem ser representados como:

$$I^{(h)} = I_0 + I_1 i + I_2 j + I_3 k, \quad (9)$$

$$F^{(h)} = F_0 + F_1 i + F_2 j + F_3 k, \quad (10)$$

onde $i = e_1$, $j = e_2$ e $k = e_{12}$

Tomando a multiplicação dentro do somatório em (8) e utilizando a tabela de multiplicação da álgebra $Cl(0, 2)$ (quaternions), temos:

$$\begin{aligned} I^{(h)}(p + S(q), c) F^{(h)}(q, c, k) &= (I_0 + I_1 i + I_2 j + I_3 k)(F_0 + F_1 i + F_2 j + F_3 k) \\ &= I_0 F_0 - I_1 F_1 - I_2 F_2 - I_3 F_3 \\ &\quad + (I_0 F_1 + I_1 F_0 + I_2 F_3 - I_3 F_2) i \\ &\quad + (I_0 F_2 - I_1 F_3 + I_2 F_0 + I_3 F_1) j \\ &\quad + (I_0 F_3 + I_1 F_2 - I_2 F_1 + I_3 F_0) k \end{aligned}$$

Então para a parte real da convolução tem-se:

$$(I^{(h)} * F^{(h)})_0(p, k) = \sum_{c=1}^C \sum_{q \in D} I^{(h)}(p + S(q), c) F^{(h)}(q, c, k) \quad (11)$$

$$\begin{aligned} &= \sum_{c=1}^C \sum_{q \in D} [I_0(p + S(q), c) F_0(q, c, k) \\ &\quad - I_1(p + S(q), c) F_1(q, c, k) \\ &\quad - I_2(p + S(q), c) F_2(q, c, k) \\ &\quad - I_3(p + S(q), c) F_3(q, c, k)] \end{aligned} \quad (12)$$

Assim, é possível determinar a parte real do resultado utilizando:

$$J_0(p, k) = g(b_0(k) + (I^{(h)} * F^{(h)})_0(p, k)) \quad (13)$$

A partir desta abordagem, é possível calcular os resultados das convoluções $(I^{(h)} * F^{(h)})_1(p, k)$, $(I^{(h)} * F^{(h)})_2(p, k)$ e $(I^{(h)} * F^{(h)})_3(p, k)$.

3.3 Camada de Pooling

A camada de pooling é frequentemente implementada como uma camada que realiza uma operação sobre uma janela deslizante, geralmente $(2, 2)$, na saída da camada de convolução anterior com o objetivo de diminuir o tamanho da imagem de entrada, o que reduz o consumo de recursos computacionais, o uso de memória e o número de parâmetros (prevenindo, dessa forma, o overfitting).

Da mesma forma que nas camadas de convolução, cada neurônio na camada de pooling está conectado à saída de um número limitado de neurônios na camada anterior, que se encontram dentro de um pequeno campo receptivo retangular. No entanto, os neurônios de pooling não têm pesos; eles apenas agregam as entradas ao utilizar uma função de agregação, como o máximo, onde apenas o valor máximo de entrada em cada campo receptivo é transmitido para a próxima camada, enquanto as outras entradas são descartadas, ou a média, onde é calculada a partir dos neurônios do campo receptivo.

Além disso, a camada de pooling ajuda a tornar os recursos aprendidos mais robustos,

tornando-os menos sensíveis a variações na posição dos objetos na imagem de entrada.

Na rede hipercomplexa, a camada de pooling realiza a operação de pooling de forma independente em cada coordenada (real, i , j ou k) dos pixels do campo receptivo.

4 Metodologia

A implementação das redes neurais foi realizada em Python, utilizando bibliotecas amplamente conhecidas e utilizadas na área de aprendizado de máquina. Vale ressaltar que a implementação da rede foi baseada em um trabalho prévio [6], fornecendo uma base sólida para o desenvolvimento deste projeto. Para operações matemáticas, foi utilizada a biblioteca NumPy, enquanto as bibliotecas Pandas e Matplotlib foram empregadas para análise e visualização de dados. A biblioteca OpenCV foi utilizada para carregar e manipular as imagens do conjunto de dados. Além disso, foram utilizadas as bibliotecas Scikit-learn e TensorFlow/Keras para criar e avaliar as redes neurais.

4.1 Descrição do Conjunto de Dados

As imagens utilizadas neste projeto foram obtidas a partir do conjunto de dados PH², proveniente do Serviço de Dermatologia do Hospital Pedro Hispano, localizado em Matosinhos, Portugal [3].

Essas imagens são representadas em formato RGB de 8 bits, com uma resolução média de 768X560 pixels. O conjunto de dados é composto por um total de 200 imagens dermatoscópicas de lesões cutâneas, distribuídas da seguinte forma: 80 manchas de pele normais, que são consideradas lesões benignas e não representam uma preocupação significativa; 80 manchas de pele anormais, que requerem atenção especial, pois têm potencial para se tornarem cancerígenas; e 40 melanomas, que são classificados como câncer de pele maligno.

É importante ressaltar que todas as imagens foram classificadas e anotadas por médicos especialistas em dermatologia. Nesse conjunto também há anotações médicas detalhadas de cada imagem. Essas anotações incluem a segmentação médica da lesão, bem como o diagnóstico clínico e histológico. Além disso, os especialistas avaliaram diversos critérios dermatoscópicos, como cores, rede de pigmentos, pontos/globulos, estrias, áreas de regressão e véu azul-esbranquiçado. No entanto, as anotações médicas não foram utilizadas neste projeto. A seguir,

são apresentadas imagens exemplificativas para cada uma das classes mencionadas.

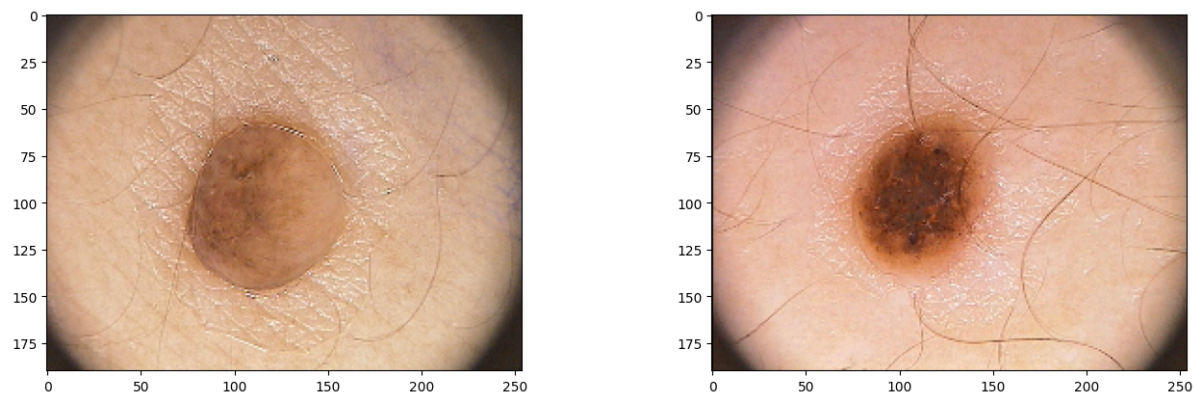


Figura 1: Manchas Normais

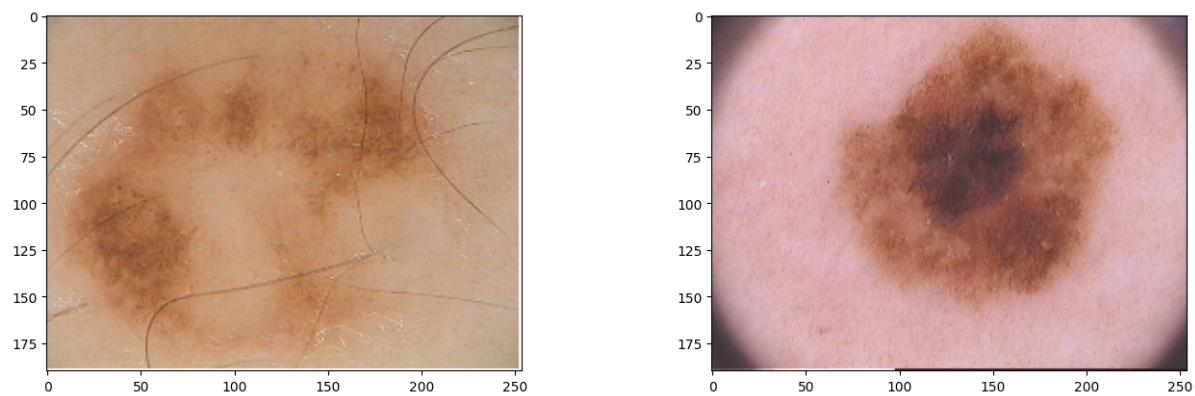


Figura 2: Manchas Anormais

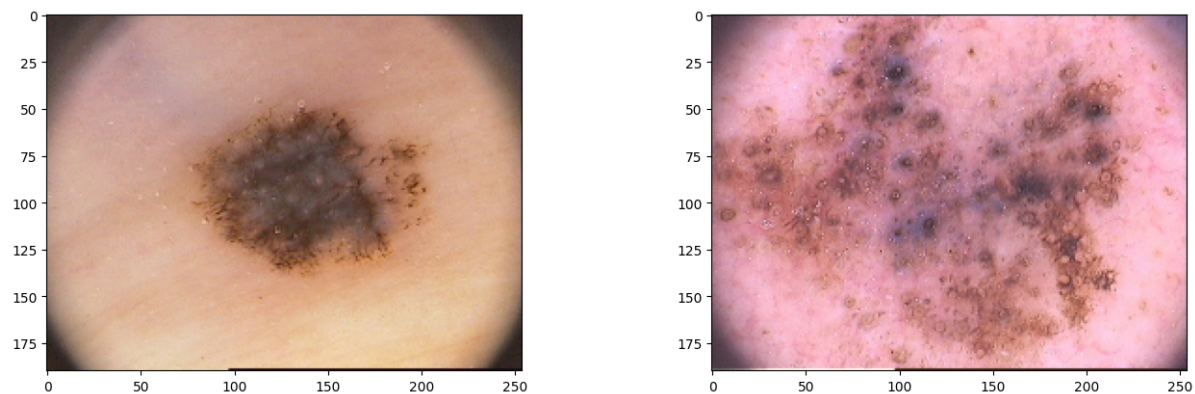


Figura 3: Melanomas

4.2 Pré-processamento

Nesta etapa, realizamos o pré-processamento das imagens com o objetivo de adequá-las ao treinamento da rede neural convolucional. Considerando que o uso das imagens originais resultaria em uma rede neural de grande porte, optamos por reduzir o tamanho das imagens, garantindo que a rede pudesse ser treinada de maneira eficiente. No entanto, foi essencial realizar essa redução com cuidado, de forma a preservar a qualidade e os detalhes importantes das imagens.

As imagens originais apresentam alturas variando entre 550 e 570 pixels, e larguras entre 750 e 770 pixels. Com base nesses valores, buscamos uma proporção de 4:3 para as dimensões das imagens reescaladas. Essa proporção foi escolhida para minimizar a distorção das imagens e preservar uma relação próxima às originais.

Além disso, consideramos a arquitetura das operações de convolução e pooling que as imagens seriam submetidas durante o treinamento da rede neural, que será vista mais a frente. É importante destacar que a camada de convolução reduzirá a altura e a largura da imagem em 2 pixels, pois serão utilizados filtros de tamanho $(3, 3)$, $stride = 1$ e *no-padding*. enquanto a camada de pooling dividirá essas dimensões por 2. Portanto, se o tamanho inicial da imagem for ímpar, a divisão resultará no número inteiro inferior mais próximo, o que pode resultar em perda de informações, especialmente nas camadas mais profundas da rede.

Levando em consideração todos esses fatores, decidimos reescalar as imagens para o tamanho final de $(190, 254)$. Essa escolha nos permite obter imagens com as seguintes dimensões após a aplicação alternada de operações de convolução e pooling:

| Operação | Magnitude da Imagem resultante |
|------------|--------------------------------|
| Inicial | (190, 254) |
| Convolução | (188, 252) |
| Pooling | (94, 128) |
| Convolução | (92, 124) |
| Pooling | (46, 62) |
| Convolução | (44, 60) |
| Pooling | (22, 30) |
| Convolução | (20, 28) |
| Pooling | (10, 14) |
| Convolução | (8, 12) |
| Pooling | (4, 6) |

(14)

4.3 Representação dos Dados

Nesta seção, descreveremos a representação dos dados utilizada no projeto, incluindo a codificação dos alvos e a estrutura dos tensores que representam as imagens.

Os alvos, representando as classes das manchas de pele, foram obtidos a partir da base de dados PH². O conjunto de classes consiste em três categorias: manchas normais, manchas anormais e melanomas. Para realizar a codificação desses alvos, utilizamos uma abordagem de vetorização, conhecida como *one-hot encoding*. Cada alvo y_i foi codificado como um vetor binário de tamanho 3, no qual o valor correspondente à classe da mancha é representado como 1 e os demais valores são 0. Por exemplo, se a mancha for normal, o vetor y_i será $[1, 0, 0]$; se for anormal, será $[0, 1, 0]$; e se for melanoma, será $[0, 0, 1]$.

As imagens foram codificadas em tensores, que são estruturas multidimensionais, adequadas para o processamento por redes neurais convolucionais. A dimensionalidade dos tensores depende do canal utilizado e da álgebra empregada.

No caso de valores reais, os tensores foram configurados com dimensões $[190, 254, 3]$, sendo que o último índice representa o número de canais. Cada canal corresponde a uma componente de cor ou característica da imagem, como Vermelho (R), Verde (G) e Azul (B), ou

Matiz (H), Saturação (S) e Valor (V).

Por outro lado, para a rede neural com valores hipercomplexos, a estrutura dos tensores foi definida com dimensões [190, 254, 4].

No caso RGB, o primeiro canal foi preenchido com valores nulos e os três canais restantes foram atribuídos aos componentes R (vermelho), G (verde) e B (azul) da imagem, respectivamente.

Já para o caso do espaço de cores HSV, cada canal do tensor foi calculado utilizando a seguinte fórmula:

$$\begin{aligned} I(p) &= (S(p) + V(p)e_1) \cdot (\cos(H(p)) + \sin(H(p))e_2), \text{ ou seja} \\ I(p) &= S(p)\cos(H(p)) + V(p)\cos(H(p))e_1 + S(p)\sin(H(p))e_2 + V(p)\sin(H(p))e_{12}, \end{aligned} \tag{15}$$

onde $H(p) \in [0, 2\pi)$ e $S(p), V(p) \in [0, 1]$, representam, respectivamente, a matiz, a saturação e o valor do pixel p .

Nessa fórmula, cada parcela representa o valor correspondente atribuído a cada canal dos tensores de valores hipercomplexos.

4.4 Arquitetura das Redes

As arquiteturas das redes neurais convolucionais, tanto com valores reais quanto com valores hipercomplexos, apresentam algumas configurações em comum:

- O tamanho dos filtros das camadas de convolução foi definido como (3,3), com um stride de 1 e utilização de pooling válido.
- A função de ativação utilizada nas camadas convolucionais foi a função ReLU.
- Foi empregado o pooling máximo com tamanho de (2,2).
- Após as camadas de convolução, foi adicionada uma camada de dropout com um fator de 0.5. Essa camada atua aleatoriamente, definindo os valores de entrada como zero com uma taxa de 0.5, com o objetivo de evitar overfitting dos dados.
- Adicionalmente, foi incluída uma camada densa de saída com 3 neurônios, representando as classes calculadas pelo modelo. A função de ativação utilizada nessa camada é a função

softmax, que é definida da seguinte forma:

$$\text{softmax}(\hat{y})_i = \frac{e^{\hat{y}_i}}{\sum_{j=1}^K e^{\hat{y}_j}} \quad (16)$$

Dessa forma, a função softmax atribui probabilidades às classes.

No caso das redes com valores reais, após realizar alguns testes, foi identificado que a arquitetura mais promissora consiste em camadas de convolução com 32, 32, 64, 64 e 128 filtros, da camada mais superficial até a mais profunda.

Já para a rede com valores hipercomplexos, as camadas de convolução foram configuradas com 8, 8, 16, 16 e 32 filtros. Esses valores foram definidos de forma a representar exatamente 1/4 do número de filtros da camada correspondente na rede com valores reais, garantindo que os tensores de saída possuam exatamente o mesmo tamanho em ambas as redes.

Essas configurações arquiteturais foram adotadas com o objetivo de otimizar o desempenho e garantir a consistência entre as redes com valores reais e valores hipercomplexos, permitindo a comparação adequada de seus resultados e análise de desempenho.

4.5 Divisão do Conjunto de Dados

A divisão do conjunto de dados foi realizada de forma apropriada, visando manter as proporções das classes nos conjuntos de treinamento, validação e teste. Dessa maneira, foram alocadas 140 imagens para o conjunto de treinamento, 20 imagens para o conjunto de validação e 40 imagens para o conjunto de teste.

Adicionalmente, é importante ressaltar que a escassez de dados de qualidade é um desafio comum nas aplicações de aprendizado de máquina na área médica, e esse estudo não foi uma exceção. Para lidar com essa limitação, foi adotada uma técnica chamada aumento de dados (data augmentation) no conjunto de treinamento e validação. Essa técnica consiste em realizar pequenas alterações nas imagens existentes, com o objetivo de aumentar o número de amostras disponíveis para treinar a rede neural.

No contexto deste estudo, o aumento de dados envolveu operações como giro horizontal e vertical, deslocamento horizontal e vertical, e pequenas variações nos valores dos pixels nos canais de cores. Essas transformações têm como propósito enriquecer o conjunto de treinamento,

gerando variações sintéticas das imagens originais. Essa abordagem visa tornar o modelo mais robusto e capacitado a lidar com variações e nuances presentes nos dados reais.

4.6 Configurações de Treinamento

Para otimizar o processo de treinamento, foi selecionado o algoritmo de otimização Adam. A escolha do Adam baseou-se em estudos anteriores que demonstraram sua eficácia e capacidade de convergência rápida.

Quanto à função de perda, foi empregada a Categorical Cross Entropy (entropia cruzada categórica), cuja equação é:

$$L(y, \hat{y}) = - \sum_{i=1}^N \sum_{j=1}^K y_{ij} \log(\hat{y}_{ij}), \quad (17)$$

onde, y_{ij} é o valor real do alvo e \hat{y}_{ij} é o valor previsto pela rede.

Essa função de perda é especialmente adequada para problemas de classificação multi-classe, como o caso em questão, no qual é necessário classificar as lesões de pele em diferentes categorias. A entropia cruzada categórica mede a divergência entre as probabilidades previstas pelo modelo e as classes reais dos dados. Ao minimizar a entropia cruzada categórica, o modelo é treinado para fazer previsões mais precisas e discriminativas, buscando minimizar a discrepância entre as probabilidades previstas e as classes verdadeiras.

O número de épocas inicialmente definido foi 500, mas foi reduzido para 300 devido à estagnação da acurácia no conjunto de validação, após um certo momento, como forma de diminuir o tempo e o custo computacional do treino. O tamanho do lote foi estabelecido como 32, determinando a quantidade de amostras processadas em cada atualização dos parâmetros durante o treinamento.

4.7 Métricas de Desempenho

Na avaliação do desempenho dos modelos, foram adotadas principalmente as métricas de acurácia e F1-Score Macro. A acurácia é uma medida que indica a proporção de previsões corretas em relação ao total de previsões feitas pelo modelo. Ela fornece uma visão geral do desempenho global do modelo.

O F1-Score é uma métrica que combina precisão e sensibilidade, a partir de uma média harmônica de ambos, para avaliar o desempenho de um modelo de classificação. O F1-Score Macro é a média aritmética dos F1-Score calculado para cada classe individualmente. Essa métrica é útil quando há um desequilíbrio entre as classes, permitindo uma avaliação mais abrangente do desempenho médio do modelo em todas as classes.

Além disso, para entender melhor as características de desempenho em cada classe, foram consideradas as métricas de Precision Micro e Recall Micro. Essas métricas fornecem uma visão mais detalhada do desempenho em nível de classe, calculando a precisão e o recall considerando todas as amostras e previsões de todas as classes. Essas métricas são particularmente úteis para identificar tendências específicas de desempenho em cada classe.

5 Resultados e Discussão

Foram realizados treinamentos de cinco repetições para cada uma das oito variações dos modelos propostos, considerando as diferentes álgebras (real e as três álgebras hipercomplexas) e os dois canais de cores (RGB e HSV). Infelizmente, devido ao elevado custo computacional do treinamento, não foi viável realizar um número maior de repetições.

Com o intuito de comparar o desempenho de cada modelo, foram plotados os gráficos abaixo apresentando a média das acurácias e F1-Score Macro obtidos.

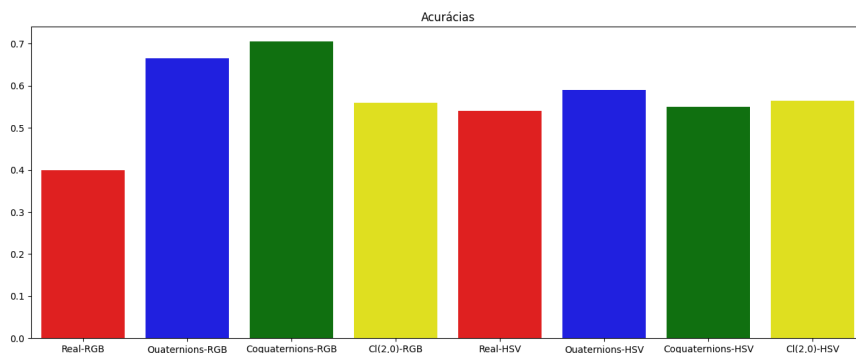


Figura 4: Gráfico das Acurácias

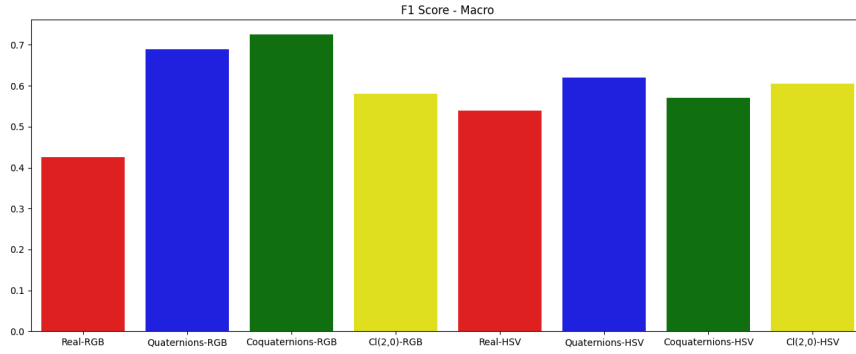


Figura 5: Gráfico dos F1 Score - Macro

5.1 Comparativo do Tamanho das Redes

A seguir, na tabela apresentada, é possível realizar uma comparação entre o tamanho das redes e suas respectivas métricas de acurácia e F1-Score Macro. Essa comparação foi feita selecionando a codificação (RGB ou HSV) que resulta o melhor desempenho para cada rede, uma vez que a codificação escolhida não interfere no tamanho da rede.

| Rede e Codificação | Número de Parâmetros | Acurácia | F1Score |
|---------------------|----------------------|----------|---------|
| Real - HSV | 148643 | 0.54 | 0.54 |
| Quaternions - RGB | 44387 | 0.665 | 0.69 |
| Coquaternions - RGB | 44387 | 0.705 | 0.725 |
| Cl(2,0) - RGB | 44387 | 0.56 | 0.58 |

(18)

5.2 Discussão sobre o desempenho e viés de cada tipo de rede

Os resultados alcançados neste estudo não se aproximam dos resultados de estado-da-arte para o problema em questão, os quais atingem uma acurácia de até 99% utilizando o mesmo conjunto de dados [2]. Isso pode ser atribuído, em parte, ao fato de que as redes desenvolvidas neste trabalho não são de grande porte em comparação com trabalhos anteriores. No entanto, é importante ressaltar que as redes com álgebra hipercomplexa apresentaram resultados superiores à rede que utilizou valores reais, especialmente a rede baseada na álgebra dos Coquaternions e codificação RGB.

Ao analisarmos separadamente as métricas de precisão, sensibilidade e F1-Score dos mo-

delos para cada classe do problema, podemos fazer algumas observações interessantes.

Para esta etapa, novamente selecionamos a melhor codificação para cada rede e calcularemos a média das métricas de precisão, recall e F1-score para cada classe. Os resultados correspondentes são apresentados na tabela abaixo.

Tabela 2: Métricas p/ cada classe: Real-HSV

| Classe | Precision | Recall | F1 Score |
|------------------|-----------|--------|----------|
| Manchas Normais | 0.56 | 0.53 | 0.54 |
| Manchas Anormais | 0.505 | 0.59 | 0.545 |
| Melanomas | 0.655 | 0.57 | 0.545 |

Tabela 3: Métricas p/ cada classe: Quaternions-RGB

| Classe | Precision | Recall | F1 Score |
|------------------|-----------|--------|----------|
| Manchas Normais | 0.58 | 0.84 | 0.685 |
| Manchas Anormais | 0.66 | 0.395 | 0.485 |
| Melanomas | 0.97 | 0.845 | 0.9 |

Tabela 4: Métricas p/ cada classe: Coquaternions-RGB

| Classe | Precision | Recall | F1 Score |
|------------------|-----------|--------|----------|
| Manchas Normais | 0.615 | 0.75 | 0.675 |
| Manchas Anormais | 0.725 | 0.515 | 0.6 |
| Melanomas | 0.88 | 0.97 | 0.915 |

Tabela 5: Métricas p/ cada classe: Cl(2,0)-RGB

| Classe | Precision | Recall | F1 Score |
|------------------|-----------|--------|----------|
| Manchas Normais | 0.515 | 0.33 | 0.4 |
| Manchas Anormais | 0.515 | 0.59 | 0.545 |
| Melanomas | 0.685 | 0.94 | 0.79 |

Nas tabelas, é possível observar que as redes hipercomplexas apresentaram um desempenho notável na identificação da classe de melanoma, cometendo poucos erros tanto de falsos

positivos quanto de falsos negativos. No entanto, para as classes de manchas normais e anormais, houve uma maior confusão na distinção entre elas, entretanto, essa é uma dificuldade que ocorre para algumas manchas, mesmo para um especialista em distinguir ambas as classes com precisão.

5.3 Análise do número de parâmetros e custo computacional

No que se refere ao custo computacional e ao tempo de execução, todas as redes apresentaram tempos semelhantes durante o treinamento, uma vez que o principal determinante desse tempo foi o número de épocas, que é invariável para todas as redes. Apesar da rede com valores reais possuir um número significativamente maior de parâmetros, as redes hipercomplexas demonstraram tempos de execução semelhantes, provavelmente devido às operações baseadas na álgebra de Clifford. No entanto, é interessante observar que, mesmo com menos parâmetros e, conseqüentemente, menor consumo de memória, as redes hipercomplexas alcançaram resultados superiores. Isso indica que a utilização dessas redes pode ser uma abordagem mais eficiente em termos de desempenho, considerando-se a relação entre número de parâmetros e resultados obtidos.

6 Conclusão

6.1 Resumo dos principais resultados e descobertas do estudo

As principais conclusões deste estudo são as seguintes:

- Os resultados revelaram a superioridade das redes hipercomplexas em relação à rede com valores reais, evidenciando maior acurácia e F1-Score Macro. Esses achados corroboram a eficácia das abordagens alternativas empregadas, especialmente ao utilizar um ferramental matemático mais avançado. Essas descobertas são promissoras e motivam a realização de estudos adicionais e testes complementares para validar e aprofundar ainda mais esses resultados.
- Por um lado, foi observada uma maior dificuldade na distinção entre as classes de manchas normais e anormais pelas redes avaliadas. Por outro lado, a classificação de mela-

noma mostrou resultados mais consistentes e precisos. Esses achados destacam a importância de desenvolver métodos mais robustos e específicos para a distinção entre manchas normais e anormais.

Esses resultados enfatizam a importância das redes hipercomplexas e reforçam a necessidade contínua de explorar abordagens inovadoras para aprimorar a classificação de lesões de pele, assim como outras aplicações potenciais da álgebra de Clifford em redes convolucionais. Essas descobertas fornecem uma base sólida para futuras investigações e incentivam a expansão do uso da álgebra de Clifford como uma ferramenta valiosa na melhoria do desempenho de redes neurais em diversas áreas de estudo.

6.2 Limitações do estudo e possíveis direções futuras de pesquisa

É importante reconhecer as limitações deste estudo e considerar possíveis direções futuras de pesquisa. Algumas limitações incluem:

- Tamanho limitado do conjunto de dados: O estudo foi realizado com um conjunto de dados limitado, o que pode afetar a representatividade e generalização dos resultados obtidos. Um conjunto de dados maior e mais diversificado poderia proporcionar uma avaliação mais abrangente do desempenho do modelo.
- Poder computacional e poucas métricas coletadas: Devido às restrições de poder computacional, foi necessário limitar o número de métricas coletadas e realizar uma análise mais simplificada dos resultados. Uma abordagem mais abrangente com um poder computacional maior permitiria a coleta de um conjunto mais extenso de métricas e uma análise mais detalhada.
- Necessidade de validação clínica: Para que as redes neurais possam ser aplicadas na prática clínica, é essencial realizar validação clínica adicional. Isso envolve testar o desempenho do modelo em casos reais, comparando-o com o diagnóstico de especialistas em dermatologia. A validação clínica forneceria uma avaliação mais precisa da eficácia do modelo no contexto médico.
- Exploração de outras arquiteturas e utilização de técnicas de aprimoramento: Futuras pesquisas podem se concentrar na exploração de outras arquiteturas de redes neurais e

técnicas para melhorar a eficiência do modelo, como *finetuning*. Essas abordagens podem ajudar a aprimorar ainda mais a precisão e o desempenho do modelo na classificação

Referências

- [1] GÉRON, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. "O'Reilly Media, Inc.", 2022.
- [2] HOSNY, K. M., KASSEM, M. A., AND FOAUD, M. M. Skin cancer classification using deep learning and transfer learning. In *2018 9th Cairo international biomedical engineering conference (CIBEC)* (2018), IEEE, pp. 90–93.
- [3] MENDONÇA, T., FERREIRA, P. M., MARQUES, J. S., MARCAL, A. R., AND ROZEIRA, J. Ph 2-a dermoscopic image database for research and benchmarking. In *2013 35th annual international conference of the IEEE engineering in medicine and biology society (EMBC)* (2013), IEEE, pp. 5437–5440.
- [4] RENAUD, P. Clifford algebras lecture notes on applications in physics, 2020.
- [5] SALIDO, J. A. A., AND RUIZ, C. Using deep learning to detect melanoma in dermoscopy images. *Int. J. Mach. Learn. Comput* 8, 1 (2018), 61–68.
- [6] VIEIRA, G., AND EDUARDO VALLE, M. Acute lymphoblastic leukemia detection using hypercomplex-valued convolutional neural networks. In *2022 International Joint Conference on Neural Networks (IJCNN)* (2022), pp. 1–8.