

Rede Neural Perceptron Multicamadas para Classificação Multiclasse de Câncer de Pele

1º Rubens Alexandre de Sousa Ferreira
Universidade Federal Rural do Semi-Árido (UFERSA)
Rio Grande do Norte, Brasil
rubens.ferreira@alunos.ufersa.edu.br

2º Rosana C. B. Rego
Departamento de Engenharias e Tecnologia
Universidade Federal Rural do Semi-Árido (UFERSA)
Rio Grande do Norte, Brasil
rosana.rego@ufersa.edu.br

Resumo—O câncer de pele é um problema relevante de saúde pública, e seu diagnóstico por computador tem sido amplamente explorado com aprendizado de máquina. Este trabalho investigou o uso de uma rede Perceptron Multicamadas (MLP) na classificação multiclasse de lesões do *dataset* HAM10000, destacando suas limitações para imagens, especialmente em cenários desbalanceados. As imagens foram redimensionadas para 64×64 , vetorizadas ($d = 12288$) e padronizadas. O modelo utilizou duas camadas densas com ReLU e saída *softmax* para 7 classes. Sete configurações foram avaliadas, variando taxa de aprendizado e número de neurônios. O melhor resultado utilizou balanceamento com SMOTE ($\alpha = 0.0001$, arquitetura 128–64) obtendo acurácia de 70,84% e F1-score ponderado de 70,32%. Apesar de capturar padrões relevantes, houve baixo desempenho em classes minoritárias e indícios de *overfitting*. Comparada a abordagens como CNNs e *Transformers*, a MLP mostrou-se limitada por ignorar a estrutura espacial, sendo adequada apenas como linha de base.

Index Terms—perceptron multicamadas, HAM10000, câncer de pele, classificação multiclasse, redes neurais artificiais

I. INTRODUÇÃO

O câncer de pele está entre as neoplasias de maior incidência em escala mundial, e sua detecção em estágios iniciais pode contribuir para a redução de atrasos no diagnóstico e para o suporte à decisão clínica [1]. Nesse cenário, a aplicação de técnicas de inteligência artificial à análise de imagens dermatoscópicas tem se consolidado como uma abordagem relevante, sobretudo com a disponibilização de bases públicas padronizadas, como o *dataset* HAM10000, formado por $N = 10015$ imagens distribuídas em $K = 7$ classes diagnósticas [2].

A classificação automática dessas imagens constitui uma tarefa complexa em razão de fatores como a semelhança entre diferentes tipos de lesões, o desbalanceamento entre as classes do conjunto de dados e a elevada variabilidade de atributos visuais, como textura, coloração, formato e bordas. Abordagens clássicas de detecção de melanoma já exploravam atributos cromáticos e características visuais como apoio ao diagnóstico, indicando a relevância da informação de cor e aparência no problema [3], [4].

Mais recentemente, arquiteturas convolucionais, estratégias de *transfer learning* e modelos baseados em *self-attention* vêm apresentando desempenho superior por conseguirem explorar relações espaciais, padrões hierárquicos e dependências globais das imagens dermatoscópicas [4], [5], [7]. Embora

arquiteturas mais recentes apresentem melhor desempenho, a MLP ainda possui importância no contexto didático das Redes Neurais Artificiais, pois sua formulação torna explícitos mecanismos como *forward propagation* (propagação direta), *backpropagation* (retropropagação do erro), uso de funções de ativação, ajuste de pesos por descida de gradiente e classificação multiclasse com *softmax* [5].

Além disso, a MLP pode ser adotada como linha de base experimental, permitindo comparar seus resultados com os de arquiteturas mais adequadas ao processamento de imagens. Nesse sentido, este artigo tem como objetivo aplicar uma rede *Multilayer Perceptron* para a classificação multiclasse de câncer de pele no *dataset* HAM10000 e, principalmente, demonstrar experimentalmente por que esse tipo de arquitetura não constitui a escolha mais adequada para esse problema. A hipótese central é que, embora a MLP seja capaz de aprender parte dos padrões discriminativos presentes nos dados, seu desempenho tende a ser inferior ao de arquiteturas que preservam as relações espaciais da imagem.

O restante deste artigo está organizado da seguinte forma: a Seção II apresenta os trabalhos relacionados; a Seção III descreve a metodologia adotada; a Seção IV discute os resultados e simulações; e, por fim, a Seção V apresenta as conclusões.

II. TRABALHOS RELACIONADOS

A literatura recente sobre classificação automática de lesões cutâneas no *dataset* HAM10000 evidencia que os melhores resultados têm sido obtidos por arquiteturas capazes de preservar e explorar a estrutura espacial das imagens, combinando pré-processamento, balanceamento, segmentação, extração de características e classificadores profundos.

Em [1], os autores apresentam uma abordagem híbrida com filtro de mediana adaptativo, segmentação por U-Net, extração de características com otimização dual e classificação com *Deep CNN*. O estudo reporta 98,76% de acurácia, 96,02% de precisão, 95,37% de *recall* e 95,32% de F1-score no HAM10000, indicando que estratégias complementares de segmentação e seleção de atributos podem ampliar significativamente a capacidade discriminativa do modelo.

Em [3], foi proposta uma arquitetura convolucional própria, utilizando imagens redimensionadas, *data augmentation*, camadas convolucionais, *max-pooling*, *dropout* e saída *softmax*. O estudo também compara a proposta com arquiteturas

consolidadas, como ResNet50 e VGG16, além de avaliá-la em diferentes bases de dados. No HAM10000, a abordagem alcança 90,00% de acurácia, 87,00% de *recall* e 85,00% de F1-score, reforçando a importância das CNNs na extração automática de padrões espaciais em imagens dermatoscópicas.

Em [5], os autores abordam um *pipeline* baseado em pré-processamento, balanceamento de classes, segmentação e classificação por *transfer learning*, avaliando arquiteturas como DenseNet169 e ResNet50. Os melhores resultados foram obtidos com DenseNet169 associada a *undersampling*, alcançando 91,20% de acurácia e 91,70% de F1-score. Esse resultado evidencia que o tratamento do desbalanceamento e a preservação das características visuais da lesão exercem influência direta sobre o desempenho final do classificador.

Além das abordagens convolucionais, modelos baseados em mecanismos de atenção também têm sido explorados na classificação de lesões cutâneas. Em [6], os autores avaliam um *Vision Transformer* (ViT) aplicado ao HAM10000 e o comparam com VGG16, ResNet50, MobileNetV2 e uma CNN customizada. Diferentemente das CNNs tradicionais, o ViT divide a imagem em *patches* (blocos), utiliza codificação posicional e mecanismos de *self-attention* para capturar relações locais e dependências globais. Segundo os resultados apresentados, o modelo atingiu 98,00% de acurácia, 98,30% de precisão, 97,80% de *recall* e 98,00% de F1-score.

Outro aspecto recorrente é o impacto do desbalanceamento entre classes. Em [8], os autores mostram que o uso de *data augmentation* melhorou o desempenho de classificadores profundos no HAM10000, com o RegNetY-320 alcançando 91,00% de acurácia, 88,10% de F1-score e ROC de 0,95 após balanceamento artificial.

Em conjunto, esses estudos indicam que o bom desempenho na classificação de lesões cutâneas depende da preservação de informações espaciais, da extração de padrões hierárquicos e do tratamento do desbalanceamento. Isso contrasta com a MLP, na qual a imagem é achatada em um vetor unidimensional, reduzindo a capacidade de explorar sua organização espacial. Assim, este trabalho adota a MLP como linha de base experimental para evidenciar suas limitações estruturais na classificação multiclasse de imagens dermatoscópicas.

III. METODOLOGIA

A. Base de dados

Neste trabalho, foi utilizado o *dataset* HAM10000, amplamente empregado em pesquisas sobre classificação automática de lesões cutâneas [2]. Esse conjunto de dados é composto por $N = 10015$ imagens dermatoscópicas de lesões pigmentadas distribuídas em $K = 7$ classes diagnósticas: melanoma (MEL - 1.113 imagens), nevos melanocíticos (NV - 6.705 imagens), carcinoma basocelular (BCC - 514 imagens) queratose benigna (BKL - 1.099 imagens), queratose actínica (AKIEC - 327 imagens), dermatofibroma (DF - 115 imagens) e lesão vascular (VASC - 142 imagens).

Seja $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ o conjunto de dados, onde $\mathbf{x}_i \in \mathbb{R}^{h \times w \times c}$ representa a i -ésima imagem RGB com altura h , largura w e $c = 3$ canais de cor, e $y_i \in \mathcal{Y} = \{0, 1, 2, 3, 4, 5, 6\}$

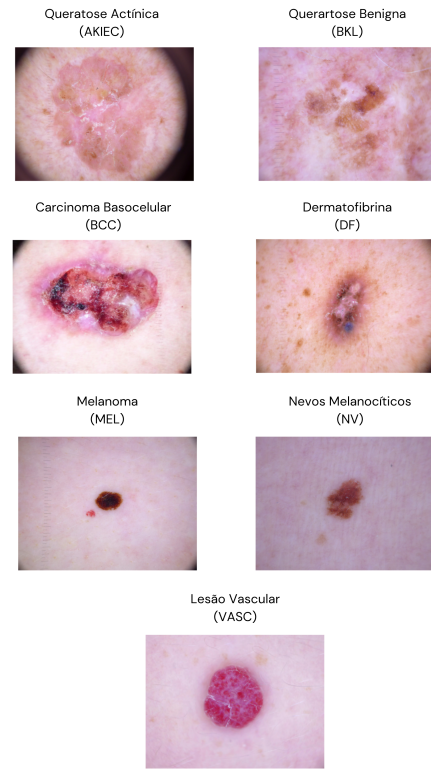


Figura 1. Exemplos ilustrativos de imagens do dataset HAM10000, com uma amostra para cada classe diagnóstica.

representa a classe diagnóstica correspondente. O objetivo é aprender uma função $f : \mathbb{R}^d \rightarrow \Delta^{K-1}$ que mapeia imagens para uma distribuição de probabilidade sobre as K classes, onde Δ^{K-1} denota o simplex de probabilidade $(K - 1)$ -dimensional.

A utilização desse *dataset* é adequada ao problema abordado, uma vez que contempla um cenário de classificação multiclasse em imagens médicas reais. No entanto, trata-se de uma base fortemente desbalanceada [8], com predominância da classe NV (aproximadamente 67% do dataset), o que torna mais complexo o processo de treinamento e avaliação dos modelos. A Fig. 1 apresenta um exemplo ilustrativo de imagem para cada uma das sete classes presentes no HAM10000.

Observa-se que as categorias exibem diferenças visuais relacionadas à coloração, textura, formato e delimitação das bordas, embora algumas delas também apresentem semelhanças relevantes. Essa variabilidade contribui para a complexidade do problema e reforça a necessidade de modelos capazes de explorar adequadamente a estrutura espacial das imagens.

B. Pré-processamento dos dados

Trata-se de um problema de classificação multiclasse supervisionada [9] [10] [11]. Seja $\mathbf{X} \in \mathbb{R}^{N \times h \times w \times c}$ o tensor completo de imagens originais. O *pipeline* de pré-processamento consiste em uma sequência de transformações $\mathcal{T} = T_4 \circ T_3 \circ T_2 \circ T_1$, onde:

Redimensionamento (T_1): Cada imagem $\mathbf{x}_i$ de tamanho arbitrário é redimensionada para dimensões fixas $h' = w' = 64$ pixels, resultando em $\mathbf{x}'_i \in \mathbb{R}^{64 \times 64 \times 3}$. Esta operação garante uniformidade dimensional e reduz o custo computacional.

Normalização (T_2): Os valores de intensidade dos pixels são normalizados para o intervalo $[0, 1]$ por meio da transformação $\mathbf{x}''_i = \mathbf{x}'_i / 255.0$, onde a divisão é aplicada elemento a elemento.

Achatamento (T_3): A imagem tridimensional $\mathbf{x}''_i \in \mathbb{R}^{64 \times 64 \times 3}$ é achatada (*flatten*) em um vetor unidimensional $\mathbf{v}_i \in \mathbb{R}^d$, onde $d = 64 \times 64 \times 3 = 12288$. Formalmente:

$$\mathbf{v}_i = \text{flatten}(\mathbf{x}''_i) \in \mathbb{R}^{12288} \quad (1)$$

Padronização (T_4): Aplicou-se o *StandardScaler* para centralizar e escalar os dados. Para cada dimensão $j \in \{1, \dots, d\}$, a transformação é dada por:

$$z_{ij} = \frac{v_{ij} - \mu_j}{\sigma_j} \quad (2)$$

onde $\mu_j = \frac{1}{N_{\text{train}}} \sum_{i \in \mathcal{D}_{\text{train}}} v_{ij}$ e $\sigma_j = \sqrt{\frac{1}{N_{\text{train}}} \sum_{i \in \mathcal{D}_{\text{train}}} (v_{ij} - \mu_j)^2}$ são a média e o desvio padrão calculados apenas sobre o conjunto de treinamento.

A divisão entre os conjuntos de treino e teste foi realizada na proporção de 80% para treino ($|\mathcal{D}_{\text{train}}| = 8012$) e 20% para teste ($|\mathcal{D}_{\text{test}}| = 2003$), com *random state* fixado em 42 e estratificação pelas classes, preservando a distribuição original do problema em ambas as partições.

Embora esse fluxo permita o treinamento de uma MLP convencional, ele também introduz uma limitação importante: o achatamento da imagem elimina as relações de vizinhança entre pixels, convertendo padrões bidimensionais em sequências lineares de atributos. Em arquiteturas mais adequadas à análise de imagens, como as redes neurais convolucionais (CNNs), essa organização espacial é preservada e explorada diretamente por filtros convolucionais. Já na MLP, tais relações precisam ser inferidas de forma indireta, o que tende a comprometer a eficiência da representação.

C. Arquitetura da rede

A arquitetura adotada neste estudo foi um *Perceptron* Multicamadas totalmente conectado (*fully-connected feedforward neural network*) [7] composto por $L = 4$ camadas sequenciais. A topologia pode ser expressa formalmente como:

$$\mathbb{R}^{12288} \xrightarrow{\text{Dense}(128)} \mathbb{R}^{128} \xrightarrow{\text{Dense}(64)} \mathbb{R}^{64} \xrightarrow{\text{Dense}(7)} \mathbb{R}^7 \quad (3)$$

onde a camada de entrada recebe vetores $\mathbf{z} \in \mathbb{R}^{12288}$, a primeira camada oculta possui $n_1 = 128$ neurônios com ativação ReLU, a segunda camada oculta contém $n_2 = 64$ neurônios com ativação ReLU, e a camada de saída apresenta $n_{\text{out}} = 7$ neurônios com ativação *Softmax* para as $K = 7$ classes diagnósticas.

Para uma camada arbitrária $l \in \{1, 2, 3\}$, a saída $\mathbf{h}^{(l)}$ é computada por:

$$\mathbf{h}^{(l)} = f^{(l)} \left(\mathbf{W}^{(l)} \mathbf{h}^{(l-1)} + \mathbf{b}^{(l)} \right) \quad (4)$$

onde $\mathbf{W}^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ é a matriz de pesos, $\mathbf{b}^{(l)} \in \mathbb{R}^{n_l}$ é o vetor de vies, $f^{(l)}$ é a função de ativação da camada l , e $\mathbf{h}^{(0)} = \mathbf{z}$ representa o vetor de entrada.

Nas camadas ocultas ($l = 1, 2$), utiliza-se a função ReLU (*Rectified Linear Unit*):

$$f_{\text{ReLU}}(x) = \max(0, x) = \begin{cases} x, & \text{se } x > 0 \\ 0, & \text{caso contrário} \end{cases} \quad (5)$$

Na camada de saída ($l = 3$), aplica-se a função *Softmax* para converter os *logits* em uma distribuição de probabilidade:

$$P(y = k | \mathbf{z}) = \sigma_k(\mathbf{h}^{(3)}) = \frac{e^{h_k^{(3)}}}{\sum_{j=1}^K e^{h_j^{(3)}}}, \quad k \in \{1, \dots, 7\} \quad (6)$$

com $\sum_{k=1}^K P(y = k | \mathbf{z}) = 1$.

A função objetivo utilizada foi a entropia cruzada categórica (*categorical cross-entropy*), apropriada para classificação multiclasse:

$$\mathcal{L}(\theta) = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \sum_{k=1}^K y_{ik} \log P(y = k | \mathbf{z}_i; \theta) \quad (7)$$

onde $\mathcal{B}$ denota um *mini-batch* de treinamento, $y_{ik} \in \{0, 1\}$ é a codificação *one-hot* da classe verdadeira, e $\theta = \{\mathbf{W}^{(l)}, \mathbf{b}^{(l)}\}_{l=1}^3$ representa o conjunto completo de parâmetros treináveis.

O processo de otimização foi realizado com o algoritmo Adam (*Adaptive Moment Estimation*), que utiliza estimativas dos momentos de primeira e segunda ordem dos gradientes, combinadas com taxas de aprendizado adaptativas. Além disso, foram utilizados pesos de classe balanceados w_k , calculados inversamente proporcionais à frequência de cada classe k no conjunto de treinamento:

$$w_k = \frac{N_{\text{train}}}{K \cdot n_k} \quad (8)$$

onde n_k é o número de amostras da classe k em $\mathcal{D}_{\text{train}}$. Esta estratégia visa reduzir os efeitos do desbalanceamento do *dataset* durante o treinamento.

De forma geral, uma rede *feedforward* pode ser representada por $\mathbf{y} = f(\mathbf{x}; \theta)$, onde $\mathbf{x}$ representa o vetor de entrada, θ representa o conjunto de parâmetros ajustáveis da rede, e $\mathbf{y}$ corresponde à saída produzida pelo modelo. Redes do tipo *Multilayer Perceptron* são capazes de modelar relações não lineares por meio da combinação de camadas ocultas, funções de ativação e retropropagação do erro [6]. Entretanto, sua eficácia depende fortemente da natureza dos atributos de entrada e da preservação de informações espaciais relevantes.

D. Configuração experimental e métricas de avaliação

Foram avaliadas sete configurações experimentais, com variação sistemática da taxa de aprendizado α e da quantidade de neurônios nas camadas ocultas (n_1, n_2), a fim de analisar

o comportamento da MLP sob diferentes ajustes de hiperparâmetros. O espaço explorado foi:

$$\begin{aligned} \alpha &\in \{0.0001, 0.001, 0.01\}, \\ (n_1, n_2) &\in \{(64, 32), (64, 64), (128, 64), \\ &\quad (256, 32), (256, 64)\} \end{aligned} \quad (9)$$

Em todos os experimentos, foram mantidos fixos os seguintes hiperparâmetros: número de épocas $E = 50$, tamanho do *mini-batch* $|\mathcal{B}| = 32$, *validation split* de 20% sobre o conjunto de treino, parâmetros do otimizador Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-7}$), e função de perda *categorical cross-entropy* com pesos de classe balanceados. A Tabela I apresenta o resumo das configurações testadas e das métricas ponderadas obtidas. Após essa etapa inicial, foi realizado um experimento adicional com balanceamento do conjunto de treinamento por meio de SMOTE, utilizando a configuração de referência com taxa de aprendizado $\alpha = 0.0001$ e arquitetura (128, 64), sem uso adicional de pesos de classe, a fim de avaliar isoladamente o impacto do balanceamento sintético no desempenho do modelo.

Para a avaliação de desempenho, foram consideradas as métricas de acurácia, precisão ponderada, *recall* ponderado e F1-score ponderado. Para um problema de classificação multiclasse com K classes, seja $C \in \mathbb{N}^{K \times K}$ a matriz de confusão, onde C_{ij} representa o número de amostras da classe verdadeira i classificadas como classe j .

As métricas ponderadas são calculadas levando em consideração o suporte de cada classe no conjunto de teste. Seja n_k o número de amostras pertencentes à classe k em D_{test} , com $|D_{test}| = \sum_{k=1}^K n_k$. A acurácia global é definida como a proporção de previsões corretas.

A adoção de métricas ponderadas mostra-se apropriada neste contexto, pois considera a contribuição de cada classe de acordo com sua frequência no conjunto de teste. Em problemas desbalanceados como o HAM10000, a acurácia isoladamente pode ser insuficiente para caracterizar o desempenho do modelo, tornando relevante a análise conjunta de métricas que incorporem a distribuição real das classes na avaliação final.

E. Balanceamento com SMOTE

Após a etapa inicial de estudo de casos, foi realizado um experimento adicional de balanceamento do conjunto de treinamento por meio da técnica SMOTE (*Synthetic Minority Over-sampling Technique*). Essa etapa teve como objetivo investigar se a geração sintética de amostras para as classes minoritárias poderia reduzir os efeitos do desbalanceamento do dataset HAM10000 e melhorar o desempenho global da MLP [9].

Esse procedimento foi aplicado após a divisão estratificada entre treino e teste, preservando o conjunto de teste original para avaliação final. Após as etapas de redimensionamento, normalização, achatamento das imagens e padronização dos atributos, foi utilizado SMOTE exclusivamente sobre o conjunto de treinamento, como essa técnica atua em espaço vetorial, sua aplicação ocorreu sobre os vetores padronizados obtidos a partir das imagens achatadas.

Para essa etapa, foi adotado como configuração de referência a arquitetura com taxa de aprendizado de 0,0001 e duas camadas ocultas com 128 e 64 neurônios, por ter apresentado o melhor equilíbrio entre as métricas ponderadas na fase inicial de testes. Após o balanceamento, os rótulos foram novamente convertidos para o formato *one-hot*, e o modelo foi treinado com os mesmos hiperparâmetros gerais já definidos anteriormente, porém sem o uso adicional de pesos de classe, a fim de isolar o balanceamento sintético sobre o processo de aprendizagem.

IV. RESULTADOS E SIMULAÇÕES

A análise dos resultados mostra que a MLP foi sensível às alterações de hiperparâmetros, especialmente à taxa de aprendizado (*learning rate*). Na etapa inicial, considerando os sete casos sem balanceamento sintético, o melhor equilíbrio entre as métricas ponderadas ocorreu no caso C2, com taxa de aprendizado $\alpha = 0.0001$ e configuração topológica $(n_1, n_2) = (128, 64)$, atingindo acurácia de $\text{Acc} = 0.6944$ e F1-score ponderado de $\text{F1}_{\text{ponderado}} = 0.6953$. Contudo, após a etapa adicional de balanceamento do conjunto de treinamento com SMOTE, o melhor desempenho global do estudo ocorreu no caso 8, que manteve a mesma configuração arquitetural e alcançou acurácia de $\text{Acc} = 0.7084$ e F1-score ponderado de $\text{F1}_{\text{ponderado}} = 0.7032$. Esses valores indicam que a rede conseguiu aprender parte dos padrões discriminativos do problema e que o balanceamento sintético contribuiu positivamente para o desempenho global, embora os resultados ainda permaneçam distantes do desempenho observado em arquiteturas específicas para visão computacional.

O caso C3 foi o pior cenário experimental. Com taxa de aprendizado $\alpha = 0.01$, excessivamente alta para a escala do problema, a rede praticamente colapsou, alcançando apenas $\text{Acc} = 0.0124$ (acurácia de 1.24%) e F1-score ponderado de 0.0022. Esse comportamento indica instabilidade severa no processo de otimização, sugerindo que a magnitude dos gradientes causou oscilações excessivas nos parâmetros durante a descida de gradiente, impedindo convergência adequada para um mínimo local razoável.

Entre as configurações com taxa de aprendizado fixa em $\alpha = 0.0001$ (casos C2, C4, C5, C6 e C7), observou-se que o aumento do número de neurônios não resultou em melhoria consistente. Embora o caso C6, com arquitetura (256, 32), tenha apresentado acurácia ligeiramente superior à do caso C2 (0.6984 contra 0.6944), suas métricas de precisão ponderada e F1-score ponderado foram inferiores. Já o caso C7, com arquitetura (256, 64), também não superou o desempenho global do C2. Esses resultados indicam que ampliar a capacidade paramétrica da MLP não é suficiente para superar suas limitações em imagens, pois a arquitetura não preserva explicitamente relações espaciais locais nem explora padrões hierárquicos como ocorre em redes convolucionais.

A etapa adicional com balanceamento por SMOTE mostrou, entretanto, que o tratamento do desbalanceamento pode gerar ganhos relevantes quando combinado à configuração de referência escolhida. Mantida a arquitetura (128 – 64) e taxa

Tabela I
CONFIGURAÇÕES TESTADAS E MÉTRICAS PONDERADAS OBTIDAS

Caso	Taxa de aprendizado	Oculto 1	Oculto 2	Acurácia	Precisão ponderada	Recall ponderado	F1-score ponderado
C1	0,0010	128	64	0,5501	0,7233	0,5501	0,5968
C2	0,0001	128	64	0,6944	0,6970	0,6944	0,6953
C3	0,0100	128	64	0,0124	0,1114	0,0124	0,0022
C4	0,0001	64	64	0,6834	0,6889	0,6834	0,6859
C5	0,0001	64	32	0,6585	0,6852	0,6585	0,6700
C6	0,0001	256	32	0,6984	0,6901	0,6984	0,6940
C7	0,0001	256	64	0,6894	0,6949	0,6894	0,6916
C8 (SMOTE)	0,0001	128	64	0,7084	0,7009	0,7084	0,7032

Tabela II
DESEMPENHO POR CLASSE NO MELHOR CASO EXPERIMENTAL (C8)

Classe	Precisão	Recall	F1-score	Suporte
MEL	0,36	0,39	0,37	223
NV	0,84	0,87	0,85	1341
BCC	0,48	0,43	0,45	103
AKIEC	0,48	0,35	0,41	65
BKL	0,45	0,42	0,43	220
DF	0,38	0,26	0,31	23
VASC	0,57	0,29	0,38	28

de aprendizado $\alpha = 0.0001$, o caso C8 superou o caso C2 em todas as métricas globais ponderadas, elevando a acurácia de 0.6944 para 0.7084. Esse resultado sugere que a superamostragem sintética contribuiu para reduzir parcialmente os efeitos da predominância da classe majoritária no processo de treinamento.

A análise por classe do melhor caso experimental (caso 8, com SMOTE) também reforça essa interpretação. A classe NV obteve desempenho muito superior às demais, com precisão $P_{NV} = 0.84$ e $recall R_{NV} = 0.87$, resultando em F1-score de 0.85. Em contraste, classes menos representadas, como DF (F1 = 0.31, suporte de 23 amostras) e VASC (F1 = 0.38, suporte 28 amostras), apresentaram métricas significativamente menores. A classe BKL também permaneceu em patamar intermediário, com F1-score de 0.43 e suporte de 220 amostras.

Esse comportamento sugere que, mesmo com o balanceamento sintético do conjunto de treinamento, o modelo ainda foi influenciado pelo forte desbalanceamento do conjunto (NV representa $\approx 67\%$ dos dados) e pela dificuldade de separar categorias com aparência visual semelhante sem preservação de informação espacial. A Tabela II detalha as métricas por classe no melhor cenário experimental.

Outro aspecto relevante é o indício de *overfitting*. Nas configurações mais promissoras, as curvas de treinamento apresentaram crescimento acentuado da acurácia de treino, enquanto a acurácia de validação permaneceu em patamar consideravelmente inferior. Formalmente, observou-se uma divergência crescente entre o erro de treino $\mathcal{L}_{train}$ e o erro

de validação $\mathcal{L}_{val}$ ao longo das épocas, com $\mathcal{L}_{val} - \mathcal{L}_{train} > \delta$ para algum limiar $\delta > 0$ significativo.

Em outras palavras, a rede passou a memorizar padrões específicos do conjunto de treino com mais facilidade do que generalizá-los para exemplos não vistos. Esse efeito é coerente com o uso de um vetor de entrada de alta dimensionalidade ($d = 12288$ atributos) em uma arquitetura totalmente conectada. Com 12288 parâmetros de entrada e duas camadas densas, o número total de parâmetros treináveis da configuração 128-64 é aproximadamente:

$$\begin{aligned}
 |\theta| &= (12288 \times 128 + 128) \\
 &\quad + (128 \times 64 + 64) \\
 &\quad + (64 \times 7 + 7) \\
 &= 1,581,703 \text{ parâmetros}
 \end{aligned} \tag{10}$$

Com $|\mathcal{D}_{train}| = 8012$ amostras, a razão parâmetros/amostras é ≈ 197.5 , indicando alta capacidade de memorização. Em comparação, arquiteturas convolucionais conseguem extrair representações mais compactas explorando compartilhamento de pesos e hierarquias espaciais, reduzindo essa razão significativamente.

Do ponto de vista conceitual e quantitativo, os resultados obtidos confirmam a hipótese proposta no início do artigo. A MLP pode funcionar como um classificador de base para o HAM10000, sobretudo quando se deseja enfatizar fundamentos de redes neurais artificiais. Contudo, quando o objetivo é maximizar desempenho em classificação de imagens, a arquitetura mostra limitações claras.

Em comparação quantitativa com os trabalhos da Seção II, o melhor resultado desta pesquisa ($Acc_{C8} = 70.84\%$, $F1_{C8} = 70.32\%$) apresentou as seguintes lacunas de desempenho (*performance gaps*): em relação ao *Deep Skin* [5], observa-se $\Delta_{Acc} = 91.2\% - 70.84\% = 20.36$ pontos percentuais e $\Delta_{F1} = 91.7\% - 70.32\% = 21.38$ p.p.; em relação ao *Vision Transformer* [6], tem-se $\Delta_{Acc} = 98.0\% - 70.84\% = 27.16$ p.p. e $\Delta_{F1} = 98.0\% - 70.32\% = 27.68$ p.p.; e em relação ao *Pipeline Híbrido* [1], obtém-se $\Delta_{Acc} = 98.76\% - 70.84\% = 27.92$ p.p. e $\Delta_{F1} = 95.32\% - 70.32\% = 25.00$ p.p.

Essas diferenças sustentam a afirmação de que a MLP não é a arquitetura ideal para esse problema. Os modelos de melhor desempenho exploram informação espacial por

meio de convoluções, mecanismos de atenção ou *pipelines* de extração de características, recursos ausentes na MLP padrão.

Além das limitações da própria arquitetura, este estudo apresenta restrições metodológicas que devem ser consideradas na interpretação dos resultados. Não foram aplicadas técnicas de *data augmentation*, as quais poderiam ampliar artificialmente o conjunto de treinamento e melhorar a generalização. Também não foram realizadas etapas específicas de remoção de artefatos visuais, como pelos, bolhas ou marcações médicas, nem procedimentos de segmentação da lesão para isolamento da região de interesse ou extração prévia de características, como textura, bordas e descritores de cor. Tais estratégias são frequentemente empregadas em trabalhos de maior desempenho na classificação de lesões cutâneas.

Outra limitação metodológica refere-se à ausência de validação cruzada *k-fold*, o que impede estimar a variabilidade do desempenho da MLP em diferentes partições de dados. Ainda assim, a divisão estratificada entre treino e teste, associada ao *validation split* interno, permitiu uma análise inicial controlada do comportamento da arquitetura. Além disso, este estudo não incluiu um experimento próprio com redes neurais convolucionais. Assim, a comparação com CNNs foi realizada apenas a partir da literatura, e não por meio de um modelo treinado e avaliado nas mesmas condições da MLP. Essa comparação direta está sendo desenvolvida como continuidade da pesquisa e será incorporada em trabalhos futuros.

Dessa forma, os resultados devem ser interpretados como uma avaliação exploratória da MLP enquanto linha de base para classificação multiclasse no HAM10000. O objetivo principal não foi obter o melhor desempenho possível para o conjunto de dados, mas demonstrar as limitações estruturais de uma arquitetura totalmente conectada quando aplicada diretamente a imagens achatadas em vetores unidimensionais, reforçando a importância de modelos capazes de preservar relações espaciais, como as redes neurais convolucionais.

V. CONCLUSÕES

Este trabalho apresentou a aplicação de uma rede neural *Perceptron* Multicamadas para classificação multiclasse de câncer de pele no *dataset* HAM10000, com o objetivo de estabelecer uma linha de base experimental e demonstrar empiricamente as limitações estruturais da arquitetura para processamento de imagens. O experimento mostrou que a MLP consegue aprender padrões relevantes e produzir uma linha de base funcional, especialmente quando configurada com taxa de aprendizado baixa ($\alpha = 0.0001$) e número moderado de neurônios nas camadas ocultas (configuração 128-64). Após a etapa inicial de testes, a aplicação do balanceamento do conjunto de treinamento por meio de SMOTE permitiu obter o melhor desempenho global do estudo, com acurácia de 70.84% e F1-score ponderado de 70.32%.

Entretanto, os resultados também evidenciaram as limitações fundamentais do modelo. O achatamento da imagem em vetores unidimensionais elimina relações espaciais essenciais entre pixels adjacentes, o forte desbalanceamento das classes continua impactando o desempenho nas classes minoritárias

(apesar do uso de pesos balanceados), e o *overfitting* surge com facilidade devido ao elevado número de parâmetros (≈ 1.58 milhões) em relação ao tamanho do conjunto de treinamento. Embora o uso de SMOTE tenha contribuído para melhorar as métricas globais ponderadas, essa estratégia não foi suficiente para superar a principal limitação da MLP: a incapacidade de explorar explicitamente a organização espacial das imagens.

Como trabalhos futuros, pretende-se dar continuidade à pesquisa com a avaliação de arquiteturas convolucionais sob protocolo experimental equivalente, incorporando técnicas de *data augmentation*, regularização, validação cruzada *k-fold* e comparação entre diferentes estratégias de tratamento do desbalanceamento. Essa etapa permitirá comparar de forma mais robusta o desempenho da MLP com modelos que preservam a estrutura espacial das imagens dermatoscópicas.

USO DE INTELIGÊNCIA ARTIFICIAL

Em conformidade com as diretrizes de integridade científica e boas práticas acadêmicas recomendadas pela CAPES, os autores declaram que utilizaram ferramentas de inteligência artificial, incluindo ChatGPT, exclusivamente para auxiliar na melhoria da linguagem e legibilidade deste manuscrito. O uso dessas ferramentas foi realizado sob supervisão humana, e os autores revisaram e editaram cuidadosamente o conteúdo, assumindo total responsabilidade pelo texto final publicado.

REFERÊNCIAS

- [1] E. Gomathi, M. Jayasheela, M. Thamara, and M. Geetha, "Skin Cancer Detection Using Dual Optimization Based Deep Learning Network," *Biomedical Signal Processing and Control*, vol. 84, art. no. 104968, 2023, doi: 10.1016/j.bspc.2023.104968.
- [2] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific Data*, vol. 5, art. no. 180161, 2018.
- [3] M. Azeem, K. Kiani, T. Mansouri and N. Topping, "SkinLesNet: Classification of Skin Lesions and Detection of Melanoma Cancer Using a Novel Multi-Layer Deep Convolutional Neural Network," *Cancers*, vol. 16, no. 1, art. no. 108, 2024.
- [4] A. Kamboj, "A color-based approach for melanoma skin cancer detection," in *Proc. 2018 First Int. Conf. on Secure Cyber Computing and Communication (ICSCCC)*, IEEE, 2018, pp. 508–513, doi: 10.1109/ICSCCC.2018.8703309.
- [5] H. L. Gururaj, N. Manju, A. Nagarjun, V. N. M. Aradhya, and F. Flammini, "Deep Skin: A Deep Learning Approach for Skin Cancer Classification," *IEEE Access*, vol. 11, pp. 50205–50214, 2023, doi: 10.1109/ACCESS.2023.3274848.
- [6] K. Naveen and M. H. Khanam, "Leveraging Vision Transformers for Automated Skin Lesion Diagnosis: A Case Study on Skin Cancer MNIST: HAM10000," in *Proc. 2025 IEEE Int. Conf. on Advanced Computing Technologies (ICACT)*, 2025, doi: 10.1109/ICACT67549.2025.11351464.
- [7] R. C. Rego, "Redes Neurais Artificiais Feedforward," material da disciplina de Redes Neurais Artificiais, UFERSA, 2026. [Online]. Available: <https://roscibely.github.io/lectures/neuralnetwork/main.html>
- [8] T. M. Alam, K. Shaukat, W. A. Khan, I. A. Hameed, L. A. Almuqren, M. A. Raza, M. Aslam, and S. Luo, "An Efficient Deep Learning-Based Skin Cancer Classifier for an Imbalanced Dataset," *Diagnostics*, vol. 12, no. 9, art. no. 2115, 2022, doi: 10.3390/diagnostics12092115.
- [9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [10] M. Harrison, *Machine Learning – Guia de Referência Rápida: Trabalhando com Dados Estruturados em Python*. São Paulo: Novatec Editora, 2019.
- [11] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2009.