



**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
COLEGIADO DO CURSO DE CIÊNCIA DA COMPUTAÇÃO**

Luiz Gabriel Bandeira e Ribeiro

**Análise de transferência de aprendizado com
rótulos fracos para classificação de câncer de
pele**

Vitória, ES

2023

Luiz Gabriel Bandeira e Ribeiro

Análise de transferência de aprendizado com rótulos fracos para classificação de câncer de pele

Monografia apresentada ao Curso de Engenharia de Computação do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Universidade Federal do Espírito Santo – UFES

Centro Tecnológico

Colegiado do Curso de Engenharia da Computação

Orientador: Prof. Dr. André Georhton Cardoso Pacheco

Vitória, ES

2023

Luiz Gabriel Bandeira e Ribeiro

Análise de transferência de aprendizado com rótulos fracos para classificação de câncer de pele

Monografia apresentada ao Curso de Engenharia de Computação do Departamento de Informática da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Engenharia de Computação.

Trabalho aprovado. Vitória, ES, 19 de Julho de 2023:

**Prof. Dr. André Georgton Cardoso
Pacheco**
Orientador

Prof. Dr. Giovanni Comarella
UFES

BSc Gabriel Giorisatto De Angelo
UFES

Vitória, ES
2023

Resumo

A detecção precoce e precisa do câncer de pele é de extrema importância para o sucesso do tratamento e a melhoria dos resultados clínicos. No entanto, a escassez de dados especializados disponíveis para treinamento de modelos de aprendizado de máquina nessa área representa um desafio significativo. Neste trabalho, foi investigada a utilização de rótulos fracos como uma alternativa para reduzir a dependência de dados especializados.

Foi construído um conjunto de dados de imagens fracamente rotuladas e então, técnicas de transferência de aprendizado foram utilizadas para treinar modelos, explorando conhecimentos prévios, adquiridos com este conjunto de dados, em tarefas relacionadas.

Os resultados obtidos não se mostraram muito positivos. As métricas de desempenho utilizadas não apresentaram melhorias significativas em comparação com um modelo treinado com dados especializados. Isso indica que, para o conjunto de dados utilizado, os rótulos fracos utilizados não forneceram informações suficientes para melhorar a detecção de câncer de pele.

Conclui-se que é necessário realizar futuros estudos explorando outras abordagens de aprendizado de máquina e estratégias de rótulos fracos. Além disso, investigar a combinação de rótulos fracos com dados especializados pode ser uma direção promissora. Esses estudos adicionais são importantes para avançar o conhecimento e descobrir se os rótulos fracos podem realmente ajudar a superar as limitações impostas pela escassez de dados especializados para o treinamento de modelos para detecção de câncer de pele.

Palavras-chave: Transferência de Aprendizado, Rótulos Fracos, Câncer de Pele, Aprendizado de Máquina.

Listas de ilustrações

Figura 1 – Representação espacial das taxas ajustadas de incidência por 100 mil homens, estimadas para o ano de 2023, segundo Unidade da Federação (todas as neoplasias malignas, exceto as de pele não melanoma)	9
Figura 2 – Representação espacial das taxas ajustadas de incidência por 100 mil mulheres, estimadas para o ano de 2023, segundo Unidade da Federação (todas as neoplasias malignas, exceto as de pele não melanoma)	10
Figura 3 – A representação do modelo do neurônio perceptron (PACHECO, 2016)	14
Figura 4 – A representação de uma Rede Neural <i>Feedforward</i> , exemplo de uma <i>ANN</i> (PACHECO, 2015).	15
Figura 5 – A arquitetura de uma <i>CNN</i> (LECUN et al., 1998)	16
Figura 6 – O processo de convolução exemplificado em dois estados: estado inicial (topo) e estado final (abaixo). A matriz azul é a matriz de entrada, a matriz vermelha é o filtro, a caixa amarela é o viés e a matriz verde é a matriz de saída. Não há preenchimento (<i>padding</i>) e o passo (<i>stride</i>) é igual a um. Adaptado de (GERON, 2017)	18
Figura 7 – Exemplo de operação de pooling utilizando <i>max pooling</i> com filtro 2×2 e deslocamento (<i>stride</i>) de 2. O valor máximo de cada quadrado colorido 2×2 é selecionado para a matriz de saída na posição correspondente à cor do quadrado. Adaptado de (GERON, 2017)	19
Figura 8 – Exemplo de aumento de dados utilizado sobre imagens médicas (NALEPA; MARCINKIEWICZ; KAWULOK, 2019)	21
Figura 9 – Ilustração de cada tipo de partição e a validação cruzada (PACHECO, 2021).	22
Figura 10 – Exemplos de imagens presentes no <i>Reddit Dataset</i> que fogem do escopo do trabalho.	25
Figura 11 – Exemplos de imagens de câncer de pele retiradas do <i>PAD-UFES-20 Dataset</i>	26
Figura 12 – Exemplos de curva <i>ROC</i> (DRAELOS, 2019).	30
Figura 13 – Curvas ROC obtidas para o modelo Resnet com o <i>Reddit Dataset</i> . . .	37
Figura 14 – Curvas ROC obtidas para o modelo Densenet com o <i>Reddit Dataset</i> . .	37
Figura 15 – Curvas ROC obtidas para o modelo VGGnet com o <i>Reddit Dataset</i> . .	38
Figura 16 – Curvas <i>ROC</i> obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o treinamento direto com <i>PAD-UFES-20 Dataset</i>	40
Figura 17 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o treinamento direto com <i>PAD-UFES-20 Dataset</i>	41

Figura 18 – Curvas <i>ROC</i> obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o <i>PAD-UFES-20 Dataset</i>	43
Figura 19 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o <i>PAD-UFES-20 Dataset</i>	44
Figura 20 – Curvas <i>ROC</i> obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o <i>PAD-UFES-20 Dataset</i> , sem a utilização dos pesos da <i>ImageNet</i>	46
Figura 21 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o <i>PAD-UFES-20 Dataset</i> , sem a utilização dos pesos da <i>ImageNet</i>	47

Lista de tabelas

Tabela 1 – Informações da base de dados coletadas do <i>Reddit</i>	25
Tabela 2 – Informações da base de dados PAD-UFES-20	26
Tabela 3 – Resultado do treinamento direto utilizando o <i>Reddit Dataset</i> (Com pesos da <i>ImageNet</i>).	36
Tabela 4 – Resultado do treinamento direto utilizando o <i>PAD-UFES-20 Dataset</i> (com pesos da <i>ImageNet</i>).	39
Tabela 5 – Resultado da transferência de aprendizado.	42
Tabela 6 – Resultado da transferência de aprendizado com os pesos obtidos do treinamento com o <i>Reddit Dataset</i> sem influência da <i>imageNet</i>	46

Sumário

1	INTRODUÇÃO	9
1.1	Visão geral	9
1.2	Motivação	11
1.3	Objetivos e resultados esperados	12
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	Aprendizado supervisionado e não supervisionado	13
2.1.1	Rótulos Fracos	14
2.2	Rede neural artificial	14
2.3	Rede neural convolutiva	16
2.3.1	Camada de convolução	17
2.3.2	Camada de ativação	18
2.3.3	Camada de pooling	18
2.3.4	Camada totalmente conectada	19
2.4	Transferência de aprendizado	20
2.5	Overfitting	21
2.6	Aumento de dados	21
2.7	Validação Cruzada	22
3	METODOLOGIA	23
3.1	Reddit Dataset	23
3.2	PAD-UFES-20 Dataset	26
3.3	Transferência de aprendizado	27
3.4	Métricas	27
3.4.1	Acurácia Balanceada	28
3.4.2	<i>Recall, Precision e F1-Score</i>	28
3.4.3	Curva ROC	29
3.4.4	<i>AUC</i>	30
3.4.5	Matriz de Confusão	31
4	RESULTADOS EXPERIMENTAIS	33
4.1	Pré-Processamento	33
4.2	Treinamento	34
4.3	Software e Hardware utilizados	35
4.4	Resultados e Discussões	35
4.4.1	Treinamento utilizando o <i>Reddit Dataset</i>	36

4.4.2	Treinamento utilizando o <i>PAD-UFES-20 Dataset</i>	39
4.4.3	Análise da transferência de aprendizado	41
4.4.4	Reddit x ImageNet	45
5	CONCLUSÃO	49
	REFERÊNCIAS	51

1 Introdução

1.1 Visão geral

O câncer é o principal problema de saúde pública em todo o mundo, representando uma das principais causas de morte e constituindo uma das principais barreiras para o aumento da expectativa de vida global. Em muitos países, é a principal ou segunda causa de morte prematura antes dos 70 anos. A incidência e a mortalidade por câncer estão aumentando rapidamente no cenário mundial, o que reforça ainda mais o seu impacto significativo na saúde global ([SUNG et al., 2021](#)).

O câncer de pele, em particular, destaca-se como a forma mais comum de câncer em todo o mundo, afetando anualmente milhões de pessoas. Ele ocorre devido a mutações nas células da pele, resultando em uma multiplicação descontrolada. Um dos principais fatores de risco para o desenvolvimento desse tipo de câncer é a exposição excessiva à radiação ultravioleta (UV) proveniente do sol ([ACS, 2021](#)). De acordo com estatísticas recentes, o câncer de pele representa uma preocupação crescente de saúde pública, com um aumento significativo na sua incidência. No Brasil, este é estimado como o tipo mais incidente, com uma previsão de 220 mil casos novos, representando cerca de 31,3% do total de casos de câncer no triênio de 2023 a 2025 ([INCA, 2023](#)). Esses números destacam a relevância desse tipo de câncer e a importância de medidas de prevenção, conscientização e de detecção precoce. Esta última desempenha um papel crucial na redução da morbidade e mortalidade associadas ao câncer de pele, uma vez que permite o tratamento em estágios iniciais da doença. Em muitos casos, a detecção precoce é fundamental para o sucesso do tratamento, aumentando as chances de cura ([PETRIE et al., 2019](#)).

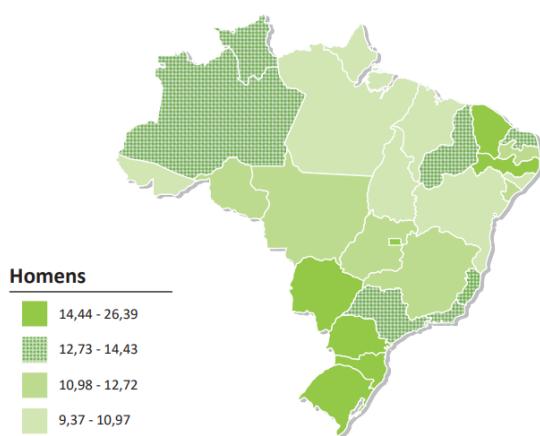


Figura 1 – Representação espacial das taxas ajustadas de incidência por 100 mil homens, estimadas para o ano de 2023, segundo Unidade da Federação (todas as neoplasias malignas, exceto as de pele não melanoma)

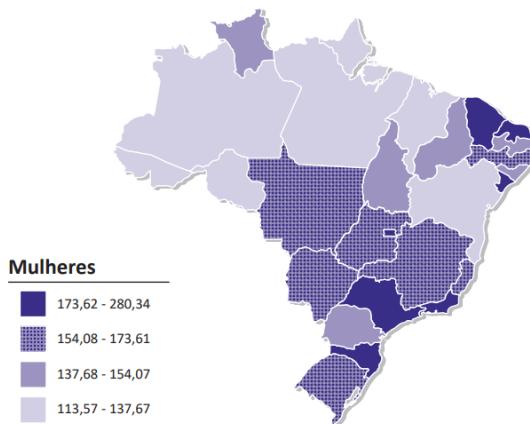


Figura 2 – Representação espacial das taxas ajustadas de incidência por 100 mil mulheres, estimadas para o ano de 2023, segundo Unidade da Federação (todas as neoplasias malignas, exceto as de pele não melanoma)

Nos últimos anos, o uso da inteligência artificial tem se mostrado cada vez mais relevante no campo da medicina, especialmente no que se refere à detecção precoce de câncer de pele (TAKIDDIN et al., 2020). Através do avanço tecnológico e da aplicação de algoritmos de aprendizado de máquina, os pesquisadores têm explorado a utilização de sistemas de diagnóstico assistido por computador (do inglês, *Computer-Aided Diagnosis — CAD*) para auxiliar no diagnóstico e na detecção de lesões cutâneas suspeitas. Estudos como (MASOOD; AL-JUMAILY, 2013) realizam análises de técnicas e algoritmos de sistemas CAD focado no diagnóstico de câncer de pele. A principal proposta desses sistemas é detectar e extraír características da lesão e ocasionalmente classificar a doença. Diante disso, mais estudos na área, como (NAHATA; SINGH, 2020) e (VIDYA; KARKI, 2020) apontam para o sucesso da utilização de algoritmos classificadores baseados em redes neurais convolucionais (do inglês, *Convolutional Neural Networks — CNN*) que empregam o aprendizado profundo (do inglês, *Deep Learning*), como também é apresentado em (DILDAR et al., 2021). Essa abordagem mostra-se extremamente promissora, uma vez que permite a extração precisa de características complexas presentes nos dados (BENGIO, 2009), de maneira a se alinhar com a proposta dos sistemas CAD.

A abordagem baseada em *deep learning* tem demonstrado sucesso, porém é fundamental ressaltar que ainda há um vasto espaço para melhorias. Uma das principais limitações encontradas reside na disponibilidade de quantidades adequadas de dados para o treinamento dos modelos. Modelos baseados em *deep learning* requerem geralmente conjuntos de dados extensos para alcançarem um desempenho satisfatório. Entretanto, na área médica, obter grandes conjuntos de dados rotulados é frequentemente um desafio devido a diversas restrições.

Dados rotulados referem-se a conjuntos de dados em que cada exemplo é acompanhado por uma etiqueta ou anotação que indica sua classe ou categoria correspondente.

No contexto médico, essas anotações podem representar diagnósticos precisos, sintomas ou características relevantes dos pacientes. Em geral, os rótulos fornecem informações valiosas para o treinamento dos modelos de aprendizado de máquina, permitindo que eles aprendam a associar corretamente os padrões presentes nos dados com suas respectivas classes.

No contexto médico, a obtenção de grandes quantidades de dados rotulados pode ser particularmente complicada. Em primeiro lugar, os dados médicos são altamente sensíveis e protegidos por regulamentos de privacidade, como a Lei Geral de Proteção de Dados (LGPD) ([BRASIL, 2018](#)). Isso impõe restrições rigorosas à coleta, armazenamento e compartilhamento dessas informações, exigindo consentimento informado dos pacientes e a garantia de sua anonimidade. Esses requisitos aumentam a complexidade e os custos envolvidos na obtenção desses conjuntos de dados. Além disso, a natureza descentralizada e fragmentada dos registros médicos dificulta a consolidação de grandes volumes de dados. Os dados médicos são armazenados em diferentes sistemas de saúde, clínicas e hospitais, frequentemente utilizando formatos e estruturas heterogêneas. A interoperabilidade e a integração desses sistemas são desafios adicionais que limitam o acesso e a disponibilidade de conjuntos de dados médicos abrangentes, como abordado em ([PASSOS, 2022](#)) e ([FORNAZIN; RACHID; NETO, 2022](#)). Além das restrições mencionadas anteriormente, a rotulagem de dados é uma tarefa que também apresenta desafios consideráveis. Ela envolve um custo significativo, depende da disponibilidade de especialistas qualificados e consome tempo considerável. A atribuição correta das etiquetas aos exemplos de dados requer perícia e conhecimento específico do domínio. A contratação de especialistas para realizar a rotulagem de dados pode ser um processo dispendioso, principalmente na área médica, onde o conhecimento técnico e a experiência são essenciais para garantir a qualidade e a confiabilidade dos rótulos atribuídos.

Em suma, a falta de dados especializados é um desafio significativo na aplicação da inteligência artificial no campo da medicina, especialmente no diagnóstico de câncer de pele. É crucial encontrar soluções para amenizar essa escassez, a fim de impulsionar o desenvolvimento e a eficácia dos sistemas de detecção e diagnóstico assistidos por computador.

1.2 Motivação

Diante dos desafios e restrições significativas citadas na seção anterior, a motivação para explorar o tema desta monografia se torna clara e pertinente. A busca por soluções inovadoras e eficientes, que permitam aproveitar rótulos fracos, isto é, anotações ou etiquetas que podem ser menos precisas ou menos especializadas em comparação com os rótulos tradicionais, e dados de menor especialização, torna-se essencial para tentar

amenizar a falta de dados especializados, expandindo a disponibilidade de conjuntos de dados médicos.

Ao buscar enxergar esses problemas como uma motivação para explorar a transferência de aprendizado utilizando rótulos fracos, é possível utilizar dados anotados de forma mais simples e menos precisa para treinar os modelos. Dessa forma, é possível aproveitar informações disponíveis em larga escala, mesmo que menos especializadas, buscando reduzir a dependência de conjuntos de dados altamente especializados e de difícil acesso.

1.3 Objetivos e resultados esperados

O objetivo geral desta monografia é explorar a viabilidade e a relevância da utilização de rótulos fracos no treinamento de redes neurais convolucionais, como as citadas em trabalhos prévios na área, com o intuito de melhorar o desempenho em tarefas específicas e amenizar a falta de dados especializados. O questionamento central é se é possível aproveitar esses dados de forma eficiente para aumentar a quantidade de dados disponíveis e preencher a lacuna dos dados especializados.

Os objetivos específicos deste trabalho visam responder as seguintes perguntas:

- A utilização de dados com rótulos fracos é efetiva para amenizar a falta de dados especializados?
- A quantidade de dados com rótulos fracos influencia positivamente o desempenho do modelo?
- A diferença no desempenho final do modelo é atribuída aos dados fracamente rotulados ou aos pesos iniciais provenientes de pré-treinos?

Em relação à última pergunta, é relevante destacar que as redes neurais frequentemente passam por um processo de pré-treinamento em conjuntos de dados extensos, como o conjunto de dados *ImageNet* ([DENG et al., 2009](#)). Essa etapa de pré-treinamento permite gerar pesos iniciais que podem ser utilizados como ponto de partida no treinamento posterior do modelo.

É esperado que as respostas a essas perguntas permitam avaliar se a abordagem de transferência de aprendizado utilizando rótulos fracos é uma estratégia efetiva para aumentar a disponibilidade de dados e aprimorar o desempenho de modelos de inteligência artificial em tarefas específicas.

A expectativa é que os resultados desta pesquisa proporcionem entendimentos valiosos sobre o potencial e as limitações do uso de rótulos fracos na transferência de aprendizado, contribuindo para o avanço da área de inteligência artificial e suas aplicações no campo médico.

2 Fundamentação Teórica

Este capítulo apresenta uma visão geral das redes neurais artificiais e redes neurais convolucionais, uma poderosa abordagem de aprendizado profundo amplamente utilizada na detecção de lesões de pele. Será abordada a arquitetura específica adotada neste estudo, explorando seus componentes fundamentais e sua aplicabilidade para o problema em questão. Além disso, serão discutidas técnicas relevantes, como *data augmentation*, *transfer learning*, aprendizagem de máquina e rótulos fracos, que desempenham um papel importante na melhoria da eficácia dos modelos de detecção. Compreender e aplicar adequadamente essas técnicas permitirão explorar o potencial das *CNN* na busca por aprimorar os resultados no contexto da detecção de lesões de pele.

2.1 Aprendizado supervisionado e não supervisionado

Os sistemas de aprendizado de máquina podem ser classificados conforme a quantidade e o tipo de supervisão que recebem durante o treinamento. Essas classificações desempenham um papel fundamental na forma como os algoritmos aprendem e como eles podem lidar com conjuntos de dados que possuem diferentes níveis de anotações ou rótulos. Existem quatro categorias principais de aprendizado de máquina: supervisionado, não supervisionado, semi-supervisionado e por reforço (GERON, 2017). Nesta seção, são abordados as duas primeiras.

No aprendizado supervisionado, os dados de treinamento são compostos por pares de entrada e saída desejada, também conhecidos como exemplos rotulados. Cada exemplo de treinamento possui uma entrada (dados de entrada) e um rótulo correspondente (saída desejada ou classe). O objetivo é aprender uma função ou um modelo que mapeie as entradas para as saídas corretas. Em outras palavras, o modelo é treinado para fazer previsões precisas com base nos rótulos fornecidos durante o treinamento (GERON, 2017).

Já o aprendizado não supervisionado lida com dados não rotulados, onde não há pares de entrada e saída correspondentes. Nessa abordagem, o modelo visa encontrar estruturas ou padrões intrínsecos nos dados, sem orientação externa. O objetivo principal é descobrir informações ocultas, agrupamentos naturais ou relações entre os dados. Em vez de ter rótulos para guiar o aprendizado, o algoritmo analisa a distribuição dos dados e procura por similaridades ou regularidades (GERON, 2017).

2.1.1 Rótulos Fracos

Os rótulos fracos, como descritos brevemente na Seção 1.2, referem-se a anotações ou etiquetas que podem ser menos precisas ou de menor especialização quando comparadas aos rótulos tradicionais, porém, que podem ainda carregar informações úteis para o treinamento dos modelos de aprendizado de máquina. Essas anotações podem ser geradas automaticamente por algoritmos, extraídas de textos não estruturados ou obtidas por meio de informações menos especializadas (SHERBURN, 2021).

A utilização de dados com rótulos fracos, da maneira feita neste trabalho, pode se enquadrar como aprendizado supervisionado, pois, mesmo que os rótulos sejam limitados ou imprecisos, ainda há a presença de algum tipo de supervisão ou orientação nas informações disponíveis. No entanto, para estudos futuros, pode ser interessante explorar o potencial do aprendizado não supervisionado no tratamento de rótulos fracos, bem como a combinação de diferentes tipos de aprendizado.

2.2 Rede neural artificial

As redes neurais artificiais (do inglês: *artificial neural networks* — *ANN*), são modelos matemáticos inspirados no funcionamento dos neurônios no cérebro humano. Elas são compostas por camadas de unidades interconectadas, chamadas de neurônios artificiais, ou **perceptrons**, que processam e transmitem informações através das conexões ponderadas entre eles (HAYKIN, 2009).

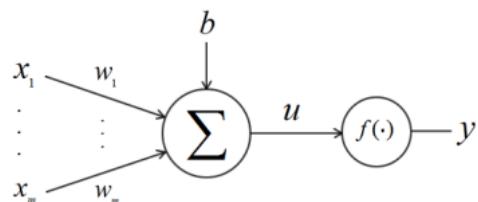


Figura 3 – A representação do modelo do neurônio perceptron (PACHECO, 2016)

Cada neurônio artificial recebe um conjunto de entradas e aplica uma função de ativação não-linear para calcular sua saída. Através do ajuste dos pesos das conexões, as redes neurais conseguem aprender padrões complexos e realizar tarefas como classificação, regressão e reconhecimento de padrões.

A saída de um Perceptron é descrita pela equação 2.1:

$$y = f \left(\sum_i^m x_i \cdot w_i + b \right) \quad (2.1)$$

na qual x são as entradas, w são os pesos, b é o bias e f é a função de ativação, componentes estes representados na Figura 3

A função de ativação mais comumente utilizada atualmente é a *ReLU* (do inglês, *Rectified Linear Unit*) descrita pela Eq. 2.2. Apesar de sua simplicidade, a função *ReLU* tem sido amplamente adotada devido à sua eficácia prática e à vantagem de ser rápida para computar (GERON, 2017).

$$f(x) = \max(0, x) \quad (2.2)$$

A arquitetura de uma *ANN* é dividida em três camadas distintas: entrada, camada oculta e saída. A camada de entrada recebe os dados alimentados na rede, enquanto a camada de saída produz o resultado do problema em questão. A camada oculta é uma parte essencial de uma *ANN*, estabelecendo conexões com a camada de entrada e com outra camada, que também pode ser uma camada oculta ou uma camada de saída.

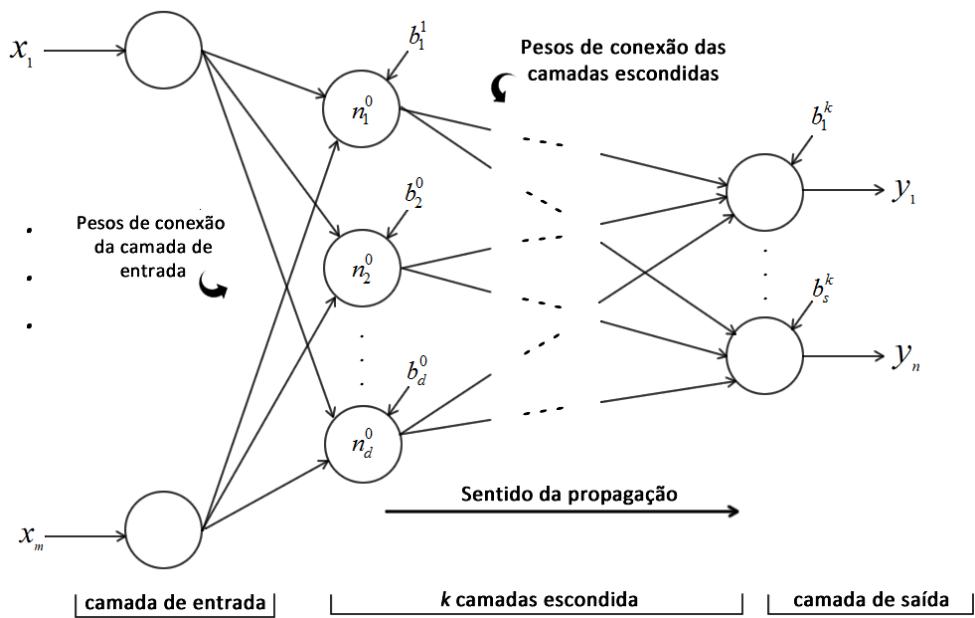


Figura 4 – A representação de uma Rede Neural *Feedforward*, exemplo de uma *ANN* (PACHECO, 2015).

Determinar o número ideal de camadas ocultas é um assunto atualmente em estudo e, normalmente, é determinado empiricamente. Por meio de experimentações e ajustes, os melhores resultados são obtidos ao escolher a quantidade adequada de camadas ocultas. Essa escolha depende da complexidade do problema, quantidade de dados disponíveis e outras considerações específicas do contexto (SHEELA; DEEPA, 2013).

O treinamento de uma rede neural envolve a apresentação de um conjunto de exemplos de treinamento, nos quais os pesos são ajustados de acordo com um algoritmo

de otimização. Durante o treinamento, a rede ajusta seus parâmetros para minimizar a diferença entre as saídas esperadas e as saídas reais, iterativamente, até alcançar um estado de convergência.

O algoritmo de otimização mais comumente utilizado, é o algoritmo de *backpropagation* (RUMELHART; HINTON; WILLIAMS, 1986). A ideia principal desse algoritmo é ajustar os pesos de conexão da rede neural visando minimizar uma função de perda usando o gradiente descendente. Uma função de perda comumente utilizada é entropia cruzada (do inglês, *cross entropy*, descrita na Eq. 2.3).

$$\text{Loss}(y, \hat{y}) = - \sum_i y_i \log(\hat{y}_i) \quad (2.3)$$

Nesta fórmula, y representa a distribuição real (rótulos verdadeiros) e \hat{y} representa a distribuição prevista pela rede neural (rótulos previstos). O símbolo \log representa o logaritmo natural.

Existem várias arquiteturas de redes neurais artificiais, como as redes neurais *feedforward*, onde a informação flui em uma única direção, das camadas de entrada para as camadas de saída. Também temos as redes neurais convolucionais, sendo especialmente eficazes em dados com estrutura espacial, como imagens, devido ao uso de filtros convolucionais para a extração de características locais.

2.3 Rede neural convolutiva

Redes Neurais Convolucionais (do inglês *Convolutional Neural Networks — CNN*) são uma classe especializada de arquiteturas de redes neurais artificiais, projetadas especialmente para lidar com dados que possuem estrutura espacial, como imagens. Elas se destacam por sua capacidade de extrair características relevantes dos dados de entrada, tornando-as extremamente eficazes em tarefas de visão computacional, como a classificação de imagens (LECUN et al., 1998).

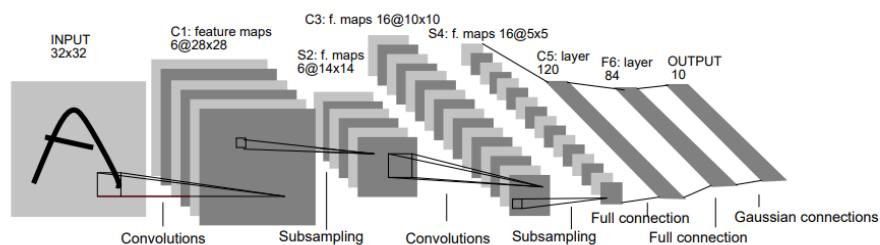


Figura 5 – A arquitetura de uma *CNN* (LECUN et al., 1998)

A principal ideia por trás das *CNNs* é a aplicação de operações de convolução nos

dados de entrada. A convolução é uma operação matemática que envolve a multiplicação de um filtro (também conhecido como *kernel*) com partes sobrepostas dos dados de entrada, seguida da soma dos resultados para produzir uma representação convolucional. Essa operação de convolução permite que as *CNNs* capturem características locais e invariantes à posição, tornando-as robustas a variações de escala, rotação e transladação nas imagens ([GERON, 2017](#)).

Uma arquitetura típica de *CNN* consiste em algumas camadas principais: de convolução, de *pooling*, de ativação e totalmente conectadas. As camadas de convolução são responsáveis por extrair características das imagens, enquanto as camadas de *pooling* reduzem a dimensionalidade das representações para diminuir a quantidade de parâmetros e processamento computacional necessário, além de também ajudar em problemas de translação e distorção de imagens. As camadas de ativação introduzem não-linearidade na rede, permitindo a aprendizagem de relações complexas entre as características. Por fim, as camadas totalmente conectadas são responsáveis por classificar as características extraídas e produzir as saídas esperadas ([GERON, 2017](#)).

Arquiteturas típicas de *CNN* empilham algumas camadas convolucionais (cada uma seguida geralmente por uma camada *ReLU*), seguidas por uma camada de *pooling*, e então mais algumas camadas convolucionais (seguidas de mais camadas *ReLU*s), seguidas por outra camada de *pooling*, e assim por diante. A imagem vai ficando cada vez menor à medida que progride pela rede, mas também geralmente fica mais profunda (ou seja, com mais mapas de características) graças às camadas convolucionais. No topo da pilha, uma rede neural *feedforward* é adicionada, composta por algumas camadas totalmente conectadas (mais camadas *ReLU*s), e a camada final gera a previsão ([LECUN et al., 1998](#)).

2.3.1 Camada de convolução

A camada de convolução é responsável por extrair características da imagem. Essa camada utiliza parâmetros que consistem em K filtros bidimensionais aprendíveis, onde cada filtro possui tamanho espacial F . Durante o processo de propagação direta (*forward pass*), cada filtro é deslizado com um passo (*stride*) de valor S pela largura e altura do volume de entrada V , é calculado, então, o produto escalar entre as entradas do filtro e a entrada em qualquer posição. Esse processo gera um mapa de ativação bidimensional que representa as respostas de cada filtro em cada posição espacial.

Em alguns casos, é conveniente adicionar zeros ao redor das bordas do volume de entrada (*padding P*), o que permite controlar o tamanho espacial dos volumes de saída. Dessa forma, a dimensão de saída D é determinada pela fórmula:

$$D = \frac{(V - F + 2P)}{S} + 1 \quad (2.4)$$

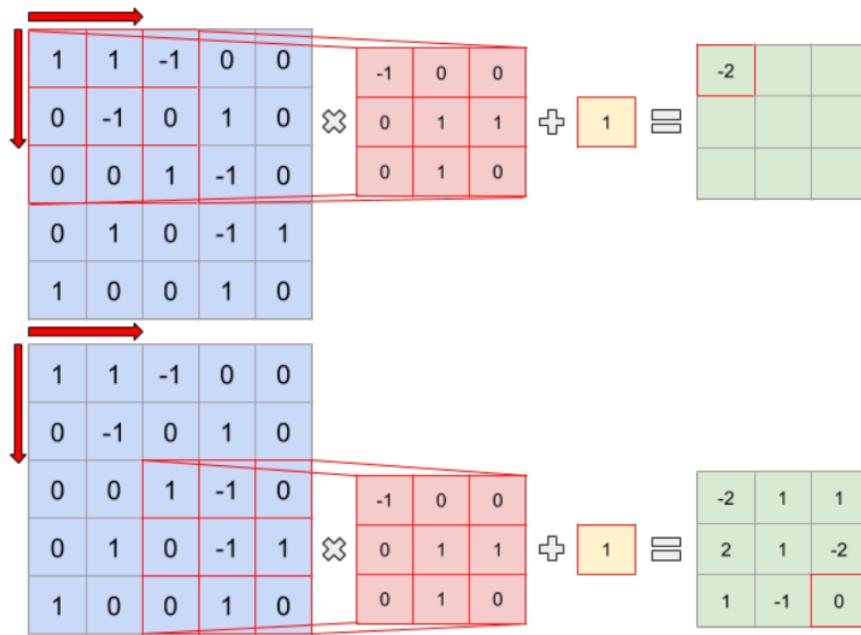


Figura 6 – O processo de convolução exemplificado em dois estados: estado inicial (topo) e estado final (abaixo). A matriz azul é a matriz de entrada, a matriz vermelha é o filtro, a caixa amarela é o viés e a matriz verde é a matriz de saída. Não há preenchimento (*padding*) e o passo (*stride*) é igual a um. Adaptado de ([GERON, 2017](#))

2.3.2 Camada de ativação

A camada de ativação é uma parte essencial das redes neurais convolucionais. Durante a camada de convolução, ocorrem operações lineares de multiplicação e soma. No entanto, é desejável introduzir não-linearidade no sistema. Para isso, é comum ter uma camada não-linear (camada de ativação) logo após cada camada de convolução. A camada de ativação consiste em uma função de ativação não linear.

Como citado na Seção 2.2, a função de ativação mais comumente utilizada é a *ReLU* e sua formulação matemática é descrita na Eq. 2.2.

A introdução da camada de ativação com a função *ReLU* após cada camada de convolução ajuda a introduzir não-linearidade no sistema, permitindo que as redes neurais convolucionais aprendam a modelar e extraírem características relevantes das imagens de forma mais eficaz ([GERON, 2017](#)).

2.3.3 Camada de pooling

Após a etapa de convolução, costuma-se incorporar uma camada de *pooling* para realizar a redução da dimensionalidade de cada mapa de características. Existem diversas abordagens de pooling espacial disponíveis, tais como a extração do máximo (*max pooling*), a média (*average pooling*) e a soma (*sum pooling*). No entanto, o pooling máximo é

amplamente utilizado por sua eficácia (GERON, 2017). Nesse método, uma região espacial é definida e o valor máximo do mapa de características retificado dentro dessa região é selecionado, formando assim uma nova matriz resultante com menor dimensionalidade, como é possível observar na Figura 7.



Figura 7 – Exemplo de operação de pooling utilizando *max pooling* com filtro 2×2 e deslocamento (*stride*) de 2. O valor máximo de cada quadrado colorido 2×2 é selecionado para a matriz de saída na posição correspondente à cor do quadrado. Adaptado de (GERON, 2017)

Essa etapa de *pooling* desempenha um papel fundamental na extração das principais informações das características identificadas durante a convolução, ao mesmo tempo, em que reduz a quantidade de dados a serem processados nas camadas subsequentes (GERON, 2017).

2.3.4 Camada totalmente conectada

A camada de ativação desempenha um papel crucial na geração de probabilidades de classes para cada entrada, especialmente em tarefas como classificação. Após a detecção de características de alto nível, é comum adicionar uma camada totalmente conectada, como pode ser observado na Figura 5. Esta camada pode ser interpretada como uma pequena *FNN* no fluxo geral de processamento da rede, com suas próprias conexões ponderadas e funções de ativação. A função da camada totalmente conectada é examinar a saída da camada anterior, pois ela possui conexões com todas as ativações anteriores, e determinar quais características estão mais correlacionadas com uma determinada classe (GERON, 2017). A saída dessa camada é um vetor N-dimensional, onde N é o número de classes do domínio, e cada valor nesse vetor representa a probabilidade de pertencer a uma determinada classe.

2.4 Transferência de aprendizado

A transferência de aprendizado (do inglês, *transfer learning*) é um conceito utilizado em técnicas de aprendizado profundo, como as redes neurais convolucionais *CNN*, quando a quantidade de dados disponíveis é limitada ou a aquisição desses dados é difícil, ou custosa, como, por exemplo, quando se trata de imagens médicas.

A ideia por trás da transferência de aprendizado é aprender a partir de um domínio de origem, onde as condições são mais favoráveis, como a disponibilidade de mais dados ou recursos computacionais, e transferir esse conhecimento para um domínio de destino ([PAN; YANG, 2010](#)). Existem diferentes tipos de transferência de aprendizado para redes neurais convolucionais, dependendo dos conjuntos de dados de origem e destino e como eles estão relacionados ([YOSINSKI et al., 2014](#)).

Quando o conjunto de dados de destino é pequeno e similar ao conjunto de dados de origem, a abordagem mais utilizada é copiar todas as camadas convolucionais treinadas do modelo de origem para o modelo de destino e congelá-las. Em seguida, a camada totalmente conectada é treinada do zero com o conjunto de dados de destino. Outra abordagem de transferência de aprendizado é copiar apenas as camadas convolucionais menos profundas do modelo de origem e congelá-las, enquanto todas as outras camadas são treinadas do zero. Isso se deve ao fato de que as primeiras camadas convolucionais retêm características mais generalizadas das imagens, como cor, textura e bordas, enquanto os parâmetros das camadas mais profundas tentam capturar características específicas do domínio de destino ([ZEILER; FERGUS, 2014](#)).

A escolha de congelar ou não algumas camadas depende do tamanho do conjunto de dados de destino e do número de parâmetros nas camadas. Se o conjunto de dados de destino for pequeno e o número de parâmetros for grande, ajustar os parâmetros pode levar ao sobreajuste, sendo melhor mantê-los congelados. Por outro lado, se o conjunto de dados de destino for grande, o sobreajuste não é um problema e ajustar os parâmetros pode levar a um desempenho melhor ([ZEILER; FERGUS, 2014](#)).

Com o crescimento do aprendizado profundo, a transferência de aprendizado tornou-se fundamental em muitas aplicações, especialmente em imagens médicas, onde o padrão atual é utilizar uma arquitetura existente projetada para conjuntos de dados de imagens naturais, como a *ImageNet*, juntamente com pesos pré-treinados correspondentes, e então ajustar o modelo com dados de imagens médicas. A transferência de aprendizado em imagens médicas permite aproveitar o conhecimento adquirido a partir de grandes conjuntos de dados de imagens naturais para melhorar o desempenho de modelos em tarefas médicas específicas. Ao iniciar com um modelo pré-treinado, que já aprendeu representações visuais gerais, e refiná-lo com dados médicos, é possível obter resultados mais precisos e eficientes no diagnóstico e análise de imagens médicas. Essa abordagem

economiza tempo e recursos, já que não é necessário treinar um modelo a partir do zero e permite que os especialistas médicos se beneficiem do conhecimento já aprendido em outras áreas da visão computacional ([RAGHU et al., 2019](#)).

2.5 Overfitting

O sobreajuste (do inglês, *overfitting*) é um fenômeno que ocorre durante o treinamento de modelos de aprendizado de máquina, no qual o modelo se ajusta muito bem aos dados de treinamento, mas tem um desempenho inferior em dados de teste ou em novos dados.

Modelos complexos, como redes neurais profundas, têm a capacidade de detectar padrões sutis nos dados. No entanto, quando o conjunto de treinamento contém ruídos ou é muito pequeno, o modelo corre o risco de detectar padrões no próprio ruído presente nos dados. Esse fenômeno se enquadra como um sobreajuste ([GERON, 2017](#)).

2.6 Aumento de dados

O aumento de dados (do inglês, *data augmentation*) é uma abordagem muito utilizada para contornar a escassez de dados de treinamento, especialmente em domínios onde a coleta de dados é limitada ou dispendiosa. Ao aplicar técnicas de aumento de dados, é possível gerar novas amostras de treinamento sintéticas, introduzindo variações nos dados existentes por meio de transformações e manipulações. Isso pode incluir rotação, espelhamento, redimensionamento, adição de ruído, entre outras técnicas.

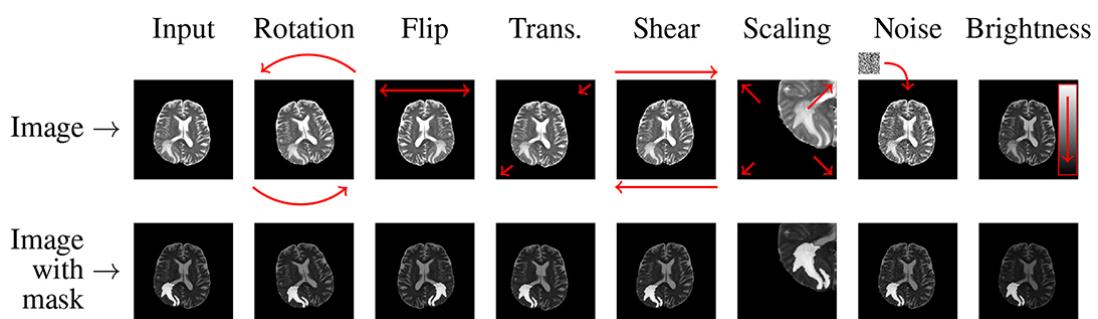


Figura 8 – Exemplo de aumento de dados utilizado sobre imagens médicas ([NALEPA; MARCINKIEWICZ; KAWULOK, 2019](#))

A principal vantagem do aumento de dados é que ele permite que o modelo aprenda com uma variedade maior de exemplos, tornando-o mais robusto e capaz de generalizar melhor para dados de teste. Além disso, o aumento de dados pode ajudar a evitar o sobreajuste, fornecendo mais variação aos dados de treinamento e evitando que o modelo se torne muito especializado em padrões específicos do conjunto de treinamento limitado

(NALEPA; MARCINKIEWICZ; KAWULOK, 2019).

2.7 Validação Cruzada

A validação cruzada (do inglês, *cross-validation*) é uma técnica utilizada na área de aprendizado de máquina para avaliar o desempenho de um modelo de forma mais robusta e confiável. Ela envolve a divisão dos dados disponíveis em conjuntos de treinamento e teste, como é possível observar na Figura 9, permitindo avaliar a capacidade do modelo de generalizar para dados não vistos.

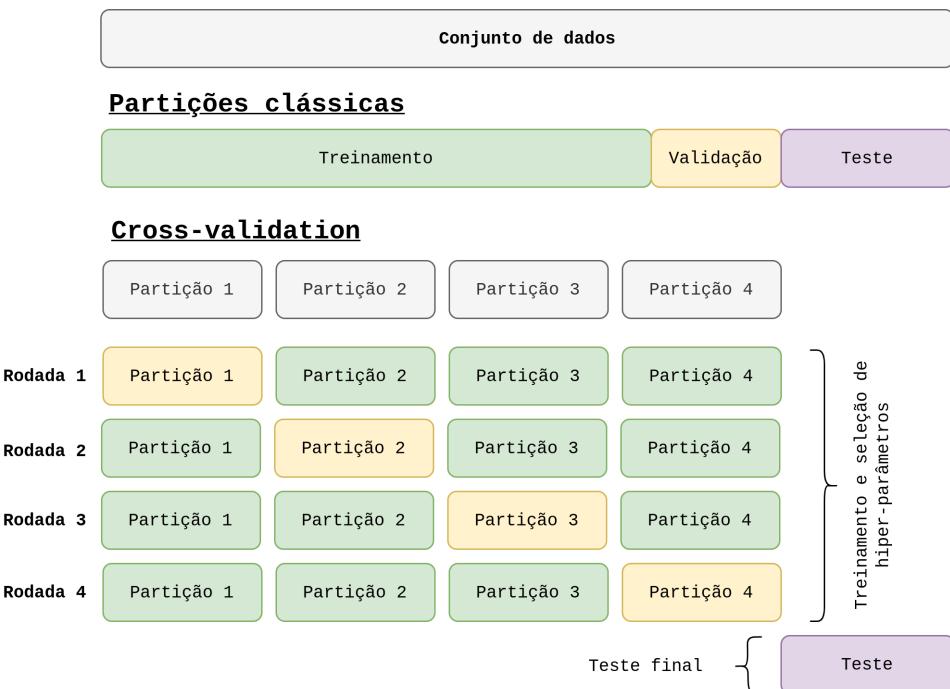


Figura 9 – Ilustração de cada tipo de partição e a validação cruzada (PACHECO, 2021).

Uma abordagem comum de validação cruzada é o *k-fold cross validation*. Nessa técnica, os dados são divididos em k subconjuntos, sendo que um subconjunto é retido como conjunto de teste e os restantes são usados como conjunto de treinamento. O processo é repetido k vezes, cada vez com um subconjunto diferente como conjunto de teste. Ao final, os resultados obtidos em cada rodada são combinados para fornecer uma estimativa mais precisa do desempenho do modelo (JAMES et al., 2013).

O *k-fold cross validation* é especialmente útil quando se tem um conjunto de dados de tamanho limitado, pois permite aproveitar ao máximo os dados disponíveis. Além disso, ele ajuda a evitar problemas como a dependência da divisão dos dados em conjuntos de treinamento e teste, o que pode levar a resultados instáveis (JAMES et al., 2013).

3 Metodologia

Este capítulo apresenta uma metodologia que visa explorar a viabilidade e a relevância da utilização de rótulos fracos no treinamento de redes neurais convolucionais para melhorar o desempenho em tarefas específicas. Como discutido anteriormente, rótulos fracos referem-se a dados que possuem imprecisões, ambiguidades ou falta de completude. No entanto, mesmo com essas limitações, busca-se investigar se é possível aproveitar esses dados de forma eficiente para aumentar a quantidade de dados disponíveis.

Neste trabalho, serão utilizados dois conjuntos de dados distintos. O primeiro conjunto de dados será o alvo, onde o objetivo é melhorar o desempenho em uma tarefa específica. Já o segundo conjunto de dados será o de estudo, contendo rótulos fracos, e será utilizado para explorar as possibilidades de transferência de aprendizado. Essa abordagem permitirá investigar como os conhecimentos adquiridos por uma rede neural convolucional em um contexto podem ser aplicados em outro.

A metodologia proposta envolve o pré-treinamento da rede neural convolucional utilizando o conjunto de dados de estudo com rótulos fracos. Essa etapa tem como objetivo explorar as características gerais dos dados e capturar informações relevantes para a tarefa de interesse. Em seguida, será realizado o ajuste fino da rede utilizando o conjunto de dados alvo, aproveitando o conhecimento prévio adquirido durante o pré-treinamento.

3.1 Reddit Dataset

Os dados utilizados neste trabalho, com rótulos fracos, foram coletados da popular rede social Reddit¹. Para isso, foi desenvolvido um *scraper*² que extraiu dados de comunidades internas da rede social, conhecidas como *subreddits*. Esses *subreddits* são divididos por tema, e para este trabalho foram utilizados os seguintes:

- 30PlusSkinCare
- Acne
- Dermatology
- Psoriasis
- Rosacea

¹ Reddit: <https://www.reddit.com>

² Reddit: https://github.com/Lab-Health/reddit_scraper

- SkincareAddiction
- SkincareAddicts
- Skincare_Addiction
- Warts
- eczema
- peeling
- popping
- skin
- skincancer

Esses *subreddits* selecionados contêm, em geral, imagens relacionadas a lesões de pele e dermatologia. É comum que os usuários dessa rede social enviem imagens para esses *subreddits* em busca de ajuda e orientação, especialmente quando têm dificuldade de acesso a hospitais e médicos devido a custos ou localização geográfica desfavorável.

É importante destacar que esse conjunto de dados é considerado fracamente rotulado, pois os rótulos são obtidos a partir dos *subreddits* aos quais as imagens foram postadas. Em outras palavras, o rótulo associado a cada imagem é o nome do *subreddit* correspondente. Essa abordagem de rotulagem pode ser considerada fraca, uma vez que os *subreddits* não representam uma classificação rigorosa das lesões de pele, mas sim uma indicação geral do tema das imagens.

Para este trabalho, o *Reddit Dataset* foi populado com 18611 imagens, sendo dispostas da seguinte forma:

Rótulo	Número de Imagens
30PlusSkinCare	670
Acne	1781
Dermatology	4026
Psoriasis	573
Rosacea	446
SkincareAddiction	1791
SkincareAddicts	1438
Skincare_Addiction	1322
Warts	2314
eczema	562
peeling	543
popping	644
skin	2072
skincancer	559
Total	18611

Tabela 1 – Informações da base de dados coletadas do *Reddit*

Devido à natureza do *dataset*, é importante ter em mente que as imagens foram capturadas por pessoas utilizando principalmente smartphones ou outros equipamentos não profissionais. Isso pode resultar em variações na qualidade das imagens, ângulos de captura e condições de iluminação. Além disso, é possível que o *dataset* contenha fotos de produtos ou outras imagens relacionadas ao tratamento das lesões de pele, além das próprias lesões.

Figura 10 – Exemplos de imagens presentes no *Reddit Dataset* que fogem do escopo do trabalho.

O uso desse *dataset* fracamente rotulado para o treinamento de pesos em algoritmos de aprendizado de máquina e inteligência artificial visa explorar técnicas de transfer learning e, futuramente, aprendizado auto-supervisionado. Compreender e lidar com a

incerteza dos rótulos é um desafio significativo, mas, ao mesmo tempo, abre oportunidades para desenvolver modelos robustos capazes de lidar com a variabilidade e a diversidade presentes nas imagens coletadas do Reddit.

3.2 PAD-UFES-20 Dataset

Para este trabalho, o *PAD-UFES-20 Dataset*, disponível em ([PACHECO et al., 2020](#)), foi escolhido como o conjunto de dados alvo para testar os efeitos da utilização de transferência de aprendizado com rótulos fracos.

O *PAD-UFES-20 Dataset* trata-se de um conjunto de dados de referência de lesões de pele, composto por imagens clínicas coletadas de dispositivos móveis e dados clínicos de pacientes contendo até 21 características. O conjunto de dados consiste em 1373 pacientes, 1641 lesões de pele e 2298 imagens para seis diagnósticos diferentes: três doenças de pele e três tipos de câncer de pele. Alguns exemplos podem ser vistos na Figura 11



Figura 11 – Exemplos de imagens de câncer de pele retiradas do *PAD-UFES-20 Dataset*

As classes podem ser observadas na tabela 2, que contém também o número de amostras para cada uma.

Rótulo	Número de Imagens
Actinic Keratosis (ACK)	543
Basal Cell Carcinoma (BCC)	442
Melanoma (MEL)	67
Nevus (NEV)	196
Squamous Cell Carcinoma (SCC)	149
Seborrheic Keratosis (SEK)	215
Total	2298

Tabela 2 – Informações da base de dados PAD-UFES-20

3.3 Transferência de aprendizado

Inicialmente, a *CNN* será treinada utilizando o *Reddit Dataset*, considerado fracamente rotulado. A ideia é que este treinamento leve a rede neural a aprender características relevantes das imagens relacionadas a lesões de pele e dermatologia presentes no conjunto de dados. Embora os rótulos sejam considerados fracos, o objetivo é analisar se a rede neural é capaz de capturar informações e padrões relevantes nas imagens mesmo com este tipo de dado.

Em seguida, a transferência de aprendizado será aplicada para o *PAD-UFES-20 Dataset*, conforme apresentado nas seções 2.4. A ideia é que a rede neural pré-treinada com o *dataset* fracamente rotulado do Reddit tenha aprendido características úteis e relevantes para a detecção e classificação de lesões de pele. Ao utilizar essa rede neural como ponto de partida, a transferência de aprendizado visa acelerar o processo de treinamento e melhorar o desempenho do modelo com o *PAD-UFES-20 Dataset*.

Uma abordagem visada neste trabalho é a comparação entre diferentes modelos e configurações. Serão treinados mais de um modelo, incluindo alguns com pré-treinamento original (por exemplo, com pesos iniciais provenientes de uma grande base de dados, como o *ImageNet*) e outros com pré-treinamento utilizando o dataset fracamente rotulado do *Reddit Dataset*. Isso permitirá analisar se os resultados alcançados são resultados do pré-treino original do modelo ou do pré-treino com o *dataset* fracamente rotulado. Essa comparação é importante para avaliar o impacto específico da utilização do *Reddit Dataset* na melhoria do desempenho do modelo em relação aos resultados obtidos apenas com o pré-treinamento tradicional.

3.4 Métricas

Para a análise, serão utilizadas artifícios de avaliação de desempenho, como acurácia balanceada, matriz de confusão, curva *ROC* e *AUC*, para avaliar o impacto do uso de rótulos fracos no treinamento da rede neural convolucional. Para compreender tais métricas, é importante primeiro definir alguns conceitos como:

- **Verdadeiros positivos (VP)**: indica a quantidade de exemplos corretamente classificados como positivos para cada classe.
- **Verdadeiros negativos (VN)**: indica a quantidade de exemplos corretamente classificados como negativos para cada classe.
- **Falsos positivos (FP)**: indica a quantidade de exemplos erroneamente classificados como positivos para cada classe.

- **Falsos negativos (FN)**: indica a quantidade de exemplos erroneamente classificados como negativos para cada classe.

Também é importante ressaltar que as definições das métricas dispostas nesta seção utilizam de exemplos binários (duas classes) para fins didáticos, porém este trabalho trata de problemas multiclasses e para isso, é utilizado o conceito de média micro e macro.

Ao calcular a média macro de uma métrica, primeiro é calculada a métrica individual para cada classe separadamente, em seguida, a média aritmética dessas métricas individuais é obtida, ignorando o desequilíbrio de classe. Isso significa que cada classe contribui igualmente para a métrica final, independentemente do tamanho de sua amostra. Já para a média micro de uma métrica, os verdadeiros positivos, falsos positivos e falsos negativos são contabilizados globalmente para todas as classes e, em seguida, a métrica é calculada usando esses valores agregados. O enfoque de média micro considera o desequilíbrio de classe, dando mais importância às classes com mais exemplos e menos importância às classes com menos exemplos (PEDREGOSA et al., 2011).

3.4.1 Acurácia Balanceada

A acurácia balanceada (do inglês, *balanced accuracy*) é uma métrica de avaliação de desempenho usada em problemas de classificação, especialmente quando os dados estão desequilibrados. Ela é uma medida que considera tanto a taxa de verdadeiros positivos quanto a taxa de verdadeiros negativos, buscando fornecer uma visão mais abrangente da capacidade de um modelo em classificar corretamente os exemplos. A fórmula para calcular a acurácia balanceada é definida pela Eq. 3.1:

$$\text{Acurácia Balanceada} = \frac{1}{2} \left(\frac{\text{VP}}{\text{VP} + \text{FN}} + \frac{\text{VN}}{\text{VN} + \text{FP}} \right) \quad (3.1)$$

Quando os dados estão desequilibrados, ou seja, quando uma classe é mais prevalente do que a outra, a acurácia tradicional (razão entre o número de exemplos classificados corretamente e o número total de exemplos) pode ser enganosa. Isso ocorre porque um modelo pode obter uma alta acurácia simplesmente prevendo sempre a classe majoritária. Sendo assim, a utilização da acurácia balanceada acaba sendo mais adequada para este trabalho em questão.

3.4.2 *Recall*, *Precision* e *F1-Score*

O *recall*, também chamado de sensibilidade, é uma métrica utilizada para avaliar a capacidade de um modelo em identificar corretamente todos os exemplos positivos em um conjunto de dados. Em outras palavras, o *recall* mede a proporção de casos positivos que o modelo consegue encontrar. Um valor alto de *recall* indica que o modelo tem uma baixa

taxa de falsos negativos, ou seja, poucos casos positivos são erroneamente classificados como negativos.

A precisão (do inglês, *precision*) é uma métrica que avalia a capacidade de um modelo em classificar corretamente os exemplos positivos entre todas as classificações positivas e negativas efetuadas pelo modelo. Em outras palavras, a precisão mede a proporção de casos corretamente classificados como positivos em relação ao total de casos classificados como positivos pelo modelo. Uma alta precisão indica que o modelo tem uma baixa taxa de falsos positivos, ou seja, poucos casos negativos são erroneamente classificados como positivos.

O *F1 Score* é uma medida que combina o *recall* e a precisão em uma única métrica, fornecendo uma visão geral do desempenho geral do modelo. É calculado usando a média harmônica do *recall* e da precisão. O *F1 Score* considera tanto os falsos positivos quanto os falsos negativos e é útil quando há um desequilíbrio entre as classes do conjunto de dados. Um valor alto de *F1 Score* indica um bom equilíbrio entre *recall* e precisão.

$$\text{Recall} = \frac{VP}{VP + FN} \quad (3.2)$$

$$\text{Precision} = \frac{VP}{VP + FP} \quad (3.3)$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.4)$$

3.4.3 Curva ROC

A curva ROC (do inglês, *Receiver Operating Characteristic*) é uma ferramenta gráfica utilizada na avaliação e análise do desempenho de modelos de classificação, especialmente em problemas de classificação binária. Ela representa a relação entre a taxa de verdadeiros positivos ((Recall)), e a taxa de falsos positivos, ou o complemento da especificidade ($1 - \text{ESP}$), em diferentes pontos de corte para a classificação do modelo (GERON, 2017).

A Especificidade é uma métrica que mede a capacidade de um modelo em identificar corretamente os casos negativos de uma classe e pode ser definida pela Eq. 4.6

$$\text{ESP} = \frac{VN}{VN + FP} \quad (3.5)$$

Em uma curva *ROC*, o eixo X representa a taxa de falsos positivos ($1 - \text{especificidade}$), enquanto o eixo Y representa a taxa de verdadeiros positivos (sensibilidade). Cada ponto na curva corresponde a um determinado limiar de classificação, ou seja, um valor

que separa as classes positiva e negativa. Ao variar esse limiar, é possível obter diferentes pontos na curva *ROC*.

Uma curva *ROC* ideal seria uma linha reta que se aproxima do canto superior esquerdo do gráfico, indicando uma alta taxa de verdadeiros positivos e uma baixa taxa de falsos positivos em todos os pontos de corte (JAMES et al., 2013). No entanto, na prática, a curva *ROC* pode assumir diferentes formas, dependendo do desempenho do modelo, como é possível observar na Figura 12.

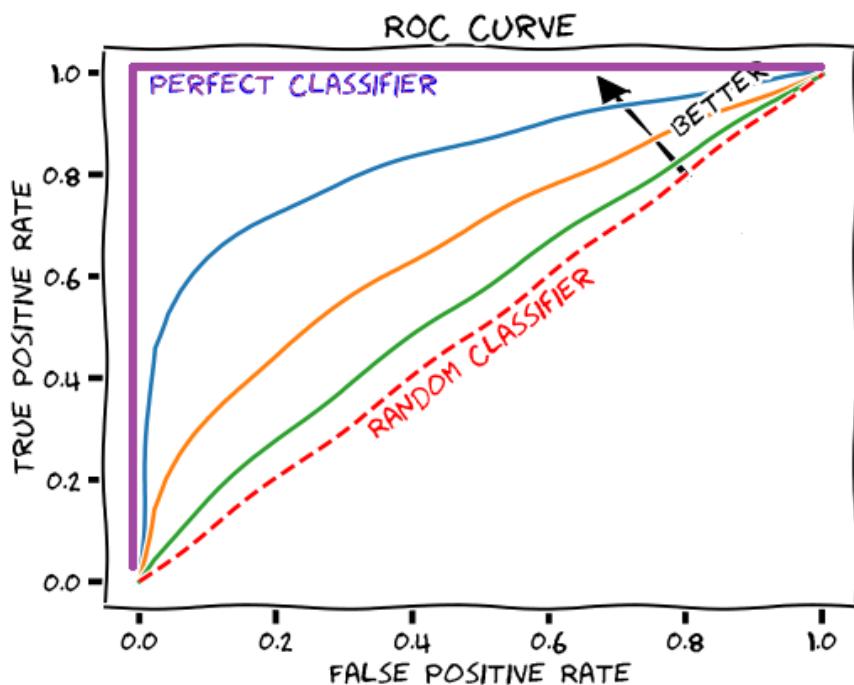


Figura 12 – Exemplos de curva *ROC* (DRAELOS, 2019).

A curva *ROC* fornece uma visão abrangente do desempenho do modelo em diferentes pontos de corte e permite escolher o limiar de classificação mais adequado, considerando as necessidades e preferências do problema em questão. Além disso, a comparação das curvas *ROC* de diferentes modelos permite determinar qual deles apresenta o melhor desempenho em termos de sensibilidade e especificidade.

3.4.4 AUC

A área Sob a Curva (do inglês, *Area Under the Curve — AUC*), é uma métrica comumente utilizada na avaliação de modelos de classificação representados pela curva *ROC*. A *AUC* mede a área total sob a curva *ROC*.

A *AUC* fornece uma medida do desempenho global do modelo em classificar corretamente as amostras em relação às classes em consideração. Ela resume a capacidade do modelo de distinguir entre exemplos positivos e negativos, independentemente do ponto

de corte do limiar de classificação utilizado (JAMES et al., 2013).

A *AUC* varia de 0 a 1, onde um valor de 0 indica um desempenho muito ruim do modelo, onde todas as previsões estão incorretas, enquanto um valor de 1 indica um desempenho perfeito do modelo, onde todas as previsões estão corretas. Um valor de 0,5 indica um desempenho aleatório, onde o modelo não possui habilidade discriminativa.

Uma *AUC* alta indica que o modelo é capaz de alcançar uma alta taxa de verdadeiros positivos (TVP) com uma baixa taxa de falsos positivos (TFP), independentemente do ponto de corte do limiar de classificação escolhido. Isso é desejável, pois significa que o modelo é capaz de fazer previsões precisas e evitar classificar erroneamente amostras negativas como positivas.

A *AUC* é uma métrica útil, especialmente quando as classes estão desbalanceadas ou quando o custo dos erros de falsos positivos e falsos negativos é diferente. Ela oferece uma medida resumida do desempenho geral do modelo em uma única pontuação, facilitando a comparação entre diferentes modelos e a seleção do melhor modelo para uma tarefa de classificação específica (JAMES et al., 2013).

3.4.5 Matriz de Confusão

A matriz de confusão é uma tabela que mostra o desempenho de um modelo de classificação em relação às classes reais de um conjunto de dados. Ela permite visualizar e avaliar as classificações corretas e incorretas feitas pelo modelo, fornecendo informações importantes sobre o desempenho em cada classe (GERON, 2017).

A matriz de confusão é construída a partir da comparação entre as classificações previstas pelo modelo e as classes verdadeiras dos exemplos. Ela organiza os resultados em uma tabela com dimensões $N \times N$, onde N é o número de classes.

A matriz de confusão é composta pelos seguintes elementos:

Esses elementos são organizados na matriz de confusão da seguinte maneira:

	Classe Positiva	Classe Negativa
Predição Positiva	Verdadeiros Positivos (VP)	Falsos Positivos (FP)
Predição Negativa	Falsos Negativos (FN)	Verdadeiros Negativos (VN)

Para calcular os elementos da matriz de confusão, é necessário comparar as classificações previstas pelo modelo com as classes verdadeiras. Cada exemplo do conjunto de dados é analisado e sua classificação é comparada com a classe verdadeira. Com base nessa comparação, os valores de VP, VN, FP e FN são atualizados.

A matriz de confusão é uma ferramenta interessante para a avaliação do desempenho de um modelo de classificação, pois fornece informações detalhadas sobre os erros e acertos em cada classe. A partir dessa matriz, diversas métricas podem ser calculadas, como

acurácia, precisão, *recall*, *F1-score*, entre outras, para avaliar o desempenho do modelo em diferentes aspectos.

4 Resultados Experimentais

Neste capítulo, apresentamos a execução dos experimentos e os resultados obtidos. Inicialmente, utilizamos o *Reddit Dataset* para o treinamento dos modelos. Em seguida, realizamos a transferência de aprendizado dos pesos obtidos para o treinamento com o *PAD-UFES-20 Dataset*. Por fim, conduzimos a análise dos resultados, focando nas métricas e nos objetivos propostos neste trabalho.

4.1 Pré-Processamento

Para o conjunto de dados de estudo, *Reddit Dataset*, o primeiro passo consiste em realizar uma auditoria das imagens visando identificar e descartar imagens problemáticas que prejudiquem o treinamento dos modelos. Essa auditoria envolve a análise visual das imagens em busca de problemas como ruídos, artefatos ou baixa qualidade, além de imagens que não se encaixam no escopo. No entanto, devido ao grande tamanho do conjunto de dados, auditar todas as imagens manualmente torna-se uma tarefa complexa e demorada. Portanto, é importante ressaltar que, mesmo com uma auditoria cuidadosa, é possível que algumas imagens problemáticas possam passar despercebidas.

Após a auditoria, as informações referentes aos dados de estudo são armazenadas em um arquivo *JSON*. Esse arquivo possui informações como o identificador de cada imagem, assim como as classes correspondentes. Para possibilitar a manipulação desses dados durante o treinamento dos modelos, é realizada a leitura do arquivo *JSON* e sua conversão para um formato adequado. Nesse processo, as informações são extraídas e organizadas em uma estrutura de dados apropriada, neste caso, um *dataframe*, que facilita o acesso e a manipulação dos dados durante as etapas subsequentes.

Em seguida, é realizada a separação das imagens em conjuntos de treino, validação e teste, que serão utilizados na técnica de *cross-validation*. A divisão dos dados em conjuntos de treino e validação é feita estratificadamente, garantindo que a distribuição das classes seja preservada em ambas as partes.

Para o conjunto de dados alvo, *PAD-UFES-20 Dataset*, o pré-processamento segue uma abordagem semelhante. No entanto, as informações referentes a esses dados são obtidas a partir de um arquivo *CSV*, que contém as características relevantes para a tarefa específica em questão.

Além do pré-processamento das informações relacionadas aos dados, também há o pré-processamento das imagens, propriamente ditas. São aplicadas transformações específicas para o pré-processamento dessas, o que inclui técnicas de aumento de dados,

como rotação, espelhamento, mudança de escala, desfoque, adição de ruído, entre outras. Essas técnicas visam aumentar a diversidade do conjunto de dados de treinamento e melhorar a capacidade do modelo em generalizar os padrões aprendidos como mencionado na Seção 2.6. Para o conjunto de dados de validação, são aplicadas transformações mais simples, como o redimensionamento da imagem e a normalização dos valores dos *pixels*.

Em resumo, o pré-processamento de imagens envolve uma etapa de auditoria (para o conjunto de dados de estudo), leitura e conversão de informações, separação em conjuntos de treino e validação, e a aplicação de técnicas de aumento de dados e transformações adequadas para cada conjunto de dados. Essas etapas permitem preparar as imagens e buscar uma melhor qualidade e consistência nos dados utilizados no treinamento e avaliação dos modelos de inteligência artificial.

4.2 Treinamento

O treinamento da rede neural foi conduzido ao longo de 150 épocas, com *early stop* definido para 15 épocas e com um *minibatch* de 30 amostras. Para o *Reddit Dataset*, utilizamos uma divisão em que 5/6, das imagens foram reservadas para o conjunto de treinamento e 1/6, para o conjunto de teste. Da mesma forma, para o *PAD-UFES-20 Dataset*, seguimos a mesma proporção de divisão.

Para ambos os conjuntos de dados, a partição de treinamento foi ainda dividida em uma proporção de 80% para treinamento e 20% para validação, utilizando-se a técnica *cross-validation* definida na Seção 2.7, com 5 *folders*. Durante o processo de treinamento, foram utilizados três modelos diferentes para comparação de resultados.

Os experimentos foram conduzidos aplicando duas metodologias distintas para investigar as questões levantadas nesta monografia. A primeira metodologia consistiu no treinamento do modelo utilizando exclusivamente as imagens do *Reddit Dataset*. Em seguida, foram analisados os resultados desse treinamento inicial. A segunda metodologia foi projetada para responder às perguntas levantadas nesta pesquisa. Nela, o modelo foi treinado inicialmente utilizando as imagens do *PAD-UFES-20 Dataset*, que serve como base para comparação direta. Em seguida, o pré-treinamento obtido na primeira metodologia, utilizando as imagens do *Reddit Dataset* com rótulos fracos, foi incorporado ao modelo por meio de transferência de aprendizado. Essa etapa visa avaliar o impacto do pré-treinamento com *Reddit Dataset* no desempenho do modelo quando treinado com o *PAD-UFES-20 Dataset*.

Conforme abordado na Seção 1.3, é comum pré-treinar redes neurais utilizando conjuntos extensos de dados, o que gera pesos iniciais que servem como ponto de partida para futuros treinamentos. No contexto deste trabalho, as redes utilizadas também foram pré-treinadas com o conjunto de dados *ImageNet*. No entanto, para explorar diferentes

cenários, foi realizado também o treinamento das redes sem utilizar os pesos adquiridos no pré-treinamento com *ImageNet*, ou seja, iniciando o treinamento com pesos aleatórios.

Para ambos os treinamentos, utilizamos um otimizador estocástico chamado SGD (*Gradiente Descendente Estocástico*), bastante utilizado na área de aprendizado de máquina (RUDER, 2016). Definimos a taxa de aprendizado como 0,0001 para ambas as metodologias de treinamento, a fim de controlar o tamanho dos passos de ajuste dos pesos da rede e para função de perda, foi utilizado no treinamento a função de entropia cruzada definida pela Eq. 2.3 na Seção 2.2.

4.3 Software e Hardware utilizados

Todo o código utilizado neste trabalho foi desenvolvido em Python, utilizando o framework PyTorch e pode ser encontrado em <https://github.com/LuizGBR/pg-ufes>. Além disso, foi utilizado o *pipeline* Raug, disponível em <https://github.com/paaatcha/raug>, para facilitar o processo de treinamento e geração de métricas.

O treinamento e os testes foram realizados em um computador contendo um Intel © Core™ i7-10700K CPU @ 3.80GHz, com 32 GB de memória RAM, placa gráfica integrada Intel © Corporation CometLake-S GT2 [UHD Graphics 630] e uma placa gráfica dedicada NVIDIA GeForce RTX 4090.

4.4 Resultados e Discussões

Nesta seção, são apresentados os resultados experimentais obtidos por meio do treinamento, seguindo a metodologia descrita nas seções seções 3.3 e 4.2. Os experimentos serão divididos em duas partes distintas. Inicialmente, serão analisados os resultados obtidos a partir do pré-treinamento da rede utilizando o *Reddit Dataset*, conforme descrito anteriormente. Para isso, serão utilizadas três arquiteturas diferentes: Resnet-50 (HE et al., 2016), Densenet-121 (HUANG et al., 2017) e VGGnet-13 (SIMONYAN; ZISSERMAN, 2014). Os resultados de cada modelo são então comparados e analisados de maneira geral.

Posteriormente, são avaliados os resultados do treinamento da rede neural com o *PAD-UFES-20 Dataset*, após a transferência de aprendizado adquirido com o *Reddit Dataset*. É importante ressaltar que para ambas as partes dos experimentos, os treinamentos utilizam os pesos da *ImageNet* como ponto de partida e forma de pré-treino. Dessa forma, também serão feitas comparações entre os resultados obtidos com e sem estes pesos.

Dessa forma, nesta seção, serão apresentados os resultados experimentais para cada etapa, a análise comparativa entre os modelos utilizados no pré-treinamento com o *Reddit Dataset* e a avaliação dos resultados do treinamento da rede com o *PAD-UFES-20 Dataset*, considerando as diferentes abordagens utilizadas.

4.4.1 Treinamento utilizando o *Reddit Dataset*

Aqui, serão apresentados e examinados os resultados do treinamento inicial da rede neural utilizando o *Reddit Dataset*, conforme detalhado na Seção 3.1. O treinamento foi realizado com cerca de 12.500 imagens, enquanto 3.200 imagens foram reservadas para validação. A fim de comparar o desempenho dos diferentes modelos utilizados, foram empregadas as seguintes métricas: acurácia balanceada, *F1-Score* e *AUC*, todas definidas na Seção 3.4.

Como já descrito, todos os experimentos foram realizados utilizando *cross-validation* com 5 *folders*. Os resultados médios, juntamente com o desvio padrão calculado, podem ser visualizados na tabela 3.

Rede	Loss	Balanced Accuracy	F1-Score	AUC
Resnet-50	2.554 ± 0.014	0.142 ± 0.012	0.134 ± 0.015	0.601 ± 0.004
Densenet-121	2.564 ± 0.016	0.143 ± 0.012	0.140 ± 0.004	0.598 ± 0.006
VGGnet-13	2.556 ± 0.020	0.134 ± 0.004	0.120 ± 0.008	0.583 ± 0.013

Tabela 3 – Resultado do treinamento direto utilizando o *Reddit Dataset* (Com pesos da *ImageNet*).

Esses resultados sugerem que, embora as redes difiram em termos de arquitetura e complexidade, elas conseguiram alcançar resultados comparáveis e bem próximos em relação às métricas avaliadas. Portanto, a escolha entre as redes pode depender de outros fatores, como eficiência computacional, disponibilidade de recursos e requisitos específicos do problema em questão.

No entanto, apesar de semelhantes, os resultados durante o treinamento com o *Reddit Dataset*, mostram um desempenho insatisfatório para as várias métricas de avaliação utilizadas. A métrica de perda indica o quanto bem o modelo se ajusta aos dados, e nesse caso, foi observada uma perda relativamente alta. Isso sugere que o modelo enfrenta dificuldades em capturar os padrões presentes no conjunto de dados.

Além disso, a acurácia balanceada, apresentou um valor baixo. Isso indica um desempenho fraco na classificação correta das classes, possivelmente tendendo a classificar a maioria das amostras em uma classe específica sem lidar adequadamente com as demais classes.

O F1-Score também mostrou um desempenho deficiente. Esse valor baixo indica muitos erros tanto em termos de falsos positivos quanto de falsos negativos, afetando a capacidade de precisão e *recall*.

A *AUC* apresentou uma média que indica uma capacidade moderada de discriminação. No entanto, ainda há bastante espaço para melhorias, pois valores mais próximos

de 1 seriam desejáveis. As curvas *ROC* referentes a cada modelo podem ser observadas nas Figuras 13, 14 e 15

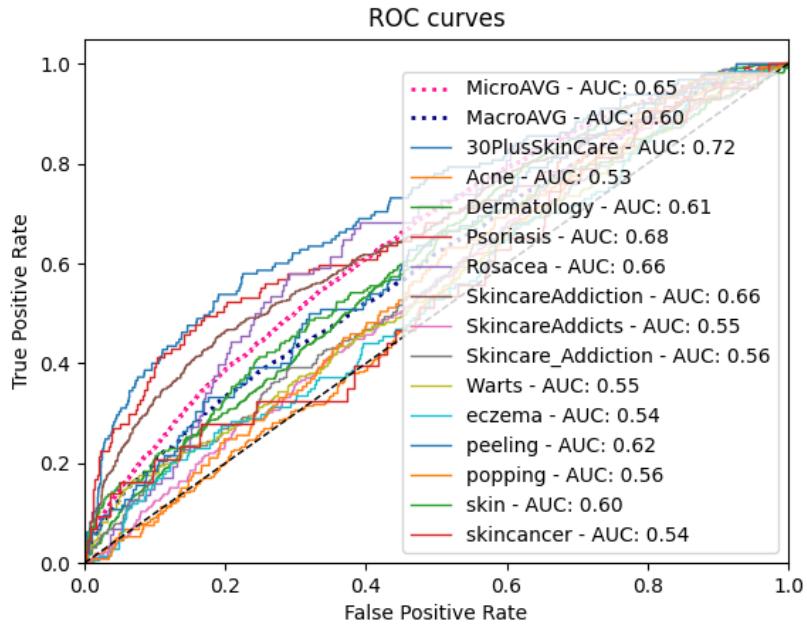


Figura 13 – Curvas ROC obtidas para o modelo Resnet com o *Reddit Dataset*

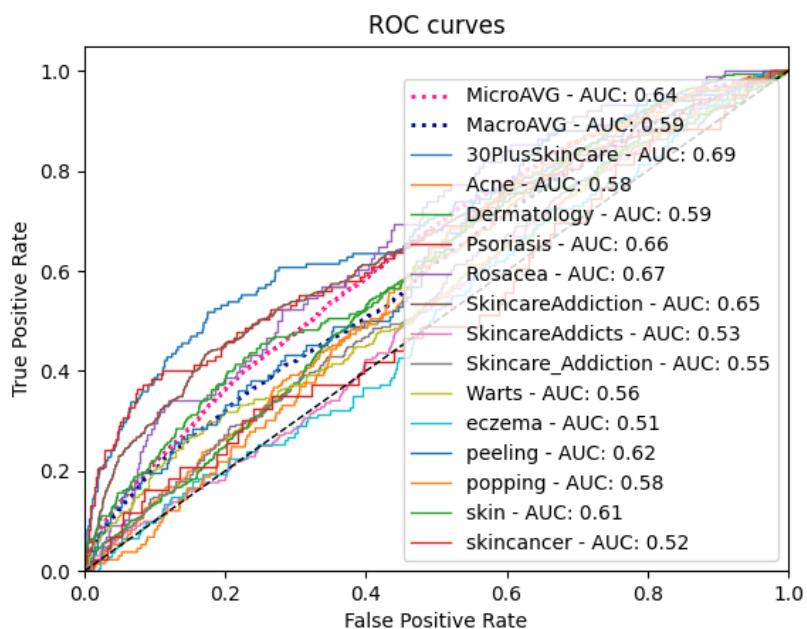


Figura 14 – Curvas ROC obtidas para o modelo DenseNet com o *Reddit Dataset*

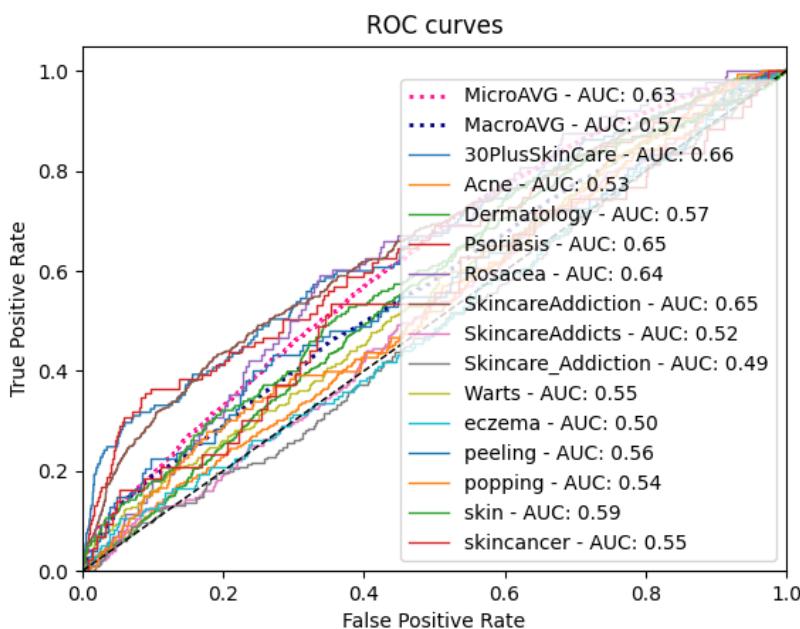


Figura 15 – Curvas ROC obtidas para o modelo VGGnet com o *Reddit Dataset*

É importante ressaltar que, comparado a outras abordagens, um desempenho inferior pode ser esperado, dadas as dificuldades mencionadas anteriormente, como o uso de rótulos fracos e as limitações do próprio conjunto de dados. A utilização de rótulos fracos pode introduzir ruído e incertezas, tornando o treinamento mais desafiador e afetando o desempenho do modelo (TURPAULT; SERIZEL; VINCENT, 2020). Estes resultados estão alinhados com a pesquisa anterior, que também constatou os efeitos negativos da introdução de ruído e viés devido aos rótulos fracos.

Além disso, o *Reddit Dataset* pode apresentar variações de qualidade e distribuição, impactando negativamente a capacidade do modelo de aprender e generalizar corretamente. É fundamental considerar essas limitações ao interpretar os resultados e avaliar o desempenho das redes neurais nesse contexto específico.

Diante dessas questões, é importante adotar estratégias adicionais para melhorar a capacidade de aprendizado e classificação. Sugestões incluem ajuste de hiperparâmetros, utilização de novas técnicas de aumento de dados e até mesmo a exploração de arquiteturas de modelos alternativas que sejam mais adequadas ao contexto do problema em questão. Outras abordagens de aprendizado de máquina como aprendizado autossupervisionado também pode ser interessante de se investigar futuramente. Essas abordagens podem ajudar a superar os desafios enfrentados pelo modelo e melhorar seu desempenho.

4.4.2 Treinamento utilizando o *PAD-UFES-20 Dataset*

A fim de realizar uma análise abrangente e uma avaliação crítica do uso de transferência de aprendizado com dados fracamente rotulados, foi conduzida uma etapa inicial de treinamento da rede neural sem a incorporação do pré-treinamento obtido a partir do *Reddit Dataset*. Para essa etapa inicial, foi selecionado um conjunto de aproximadamente 1500 imagens para o treinamento, enquanto um conjunto adicional de 400 imagens foi reservado para fins de validação. A utilização das mesmas métricas empregadas na seção anterior permitirá uma análise comparativa consistente do desempenho do modelo e fornecerá entendimentos sobre a eficácia da abordagem de transferência de aprendizado.

A tabela 4 contém os resultados médios obtidos para cada métrica proposta, juntamente dos respectivos desvios padrão.

Rede	Loss	Balanced Accuracy	F1-Score	AUC
Resnet-50	0.944 ± 0.069	0.638 ± 0.011	0.654 ± 0.025	0.903 ± 0.008
Densenet-121	0.884 ± 0.040	0.662 ± 0.028	0.664 ± 0.007	0.907 ± 0.005
VGGnet-13	0.804 ± 0.067	0.681 ± 0.026	0.686 ± 0.018	0.920 ± 0.008

Tabela 4 – Resultado do treinamento direto utilizando o *PAD-UFES-20 Dataset* (com pesos da *ImageNet*).

Os resultados obtidos são consistentes e estão alinhados com resultados obtidos em outros trabalhos ([PACHECO; KROHLING, 2020](#)). A acurácia balanceada do modelo demonstrou um desempenho razoável na classificação das classes, considerando possíveis desequilíbrios na distribuição das amostras entre elas. Além disso, o *F1-Score* revelou variações entre as classes, com algumas obtendo pontuações mais altas, indicando um bom equilíbrio entre precisão e recall nessas classes específicas. A *AUC* foi satisfatória, evidenciando a capacidade do modelo em distinguir entre as classes e apresentar um bom desempenho geral na classificação. As curvas *ROC* dos modelos podem ser visualizadas na Figura 16.

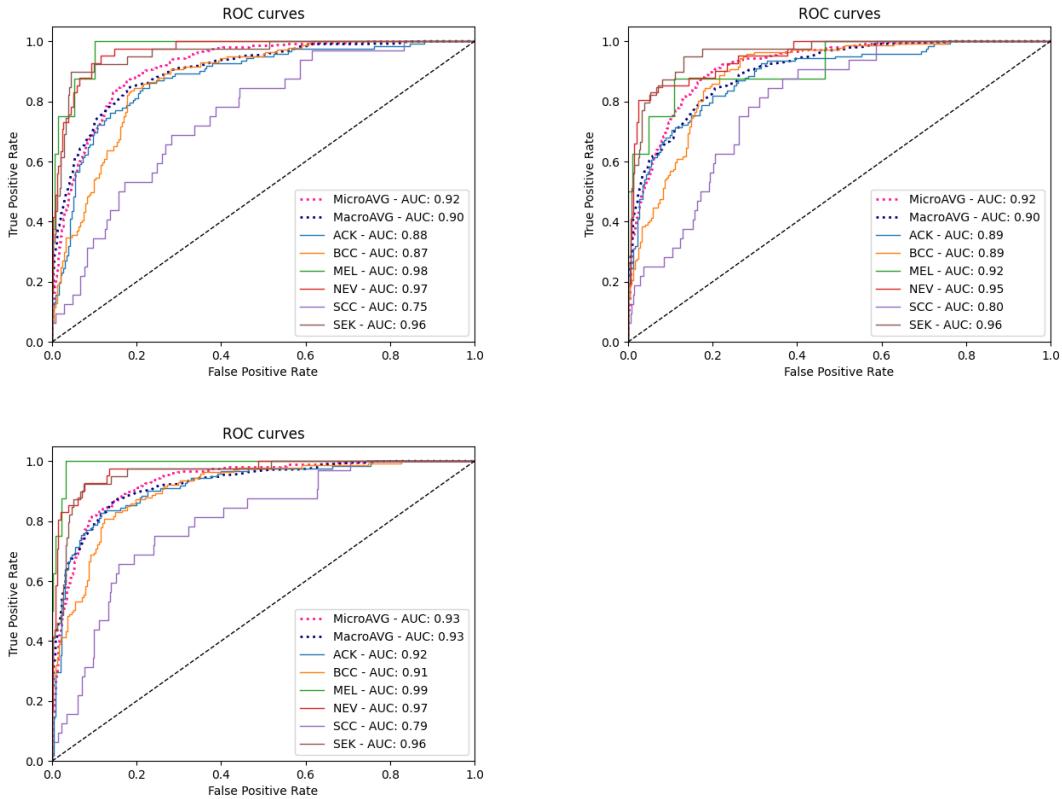


Figura 16 – Curvas *ROC* obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o treinamento direto com *PAD-UFES-20 Dataset*

Além das métricas mencionadas anteriormente, também é possível avaliar a matriz de confusão gerada durante o treinamento. A matriz de confusão fornece informações valiosas sobre o desempenho do modelo em relação à classificação das amostras em diferentes classes. Ao considerar a matriz de confusão, podemos obter entendimentos sobre o desempenho do modelo em relação a cada classe individualmente, bem como sua capacidade geral de distinguir entre as diferentes classes. As matrizes obtidas para cada modelo podem ser visualizadas na Figura 17.

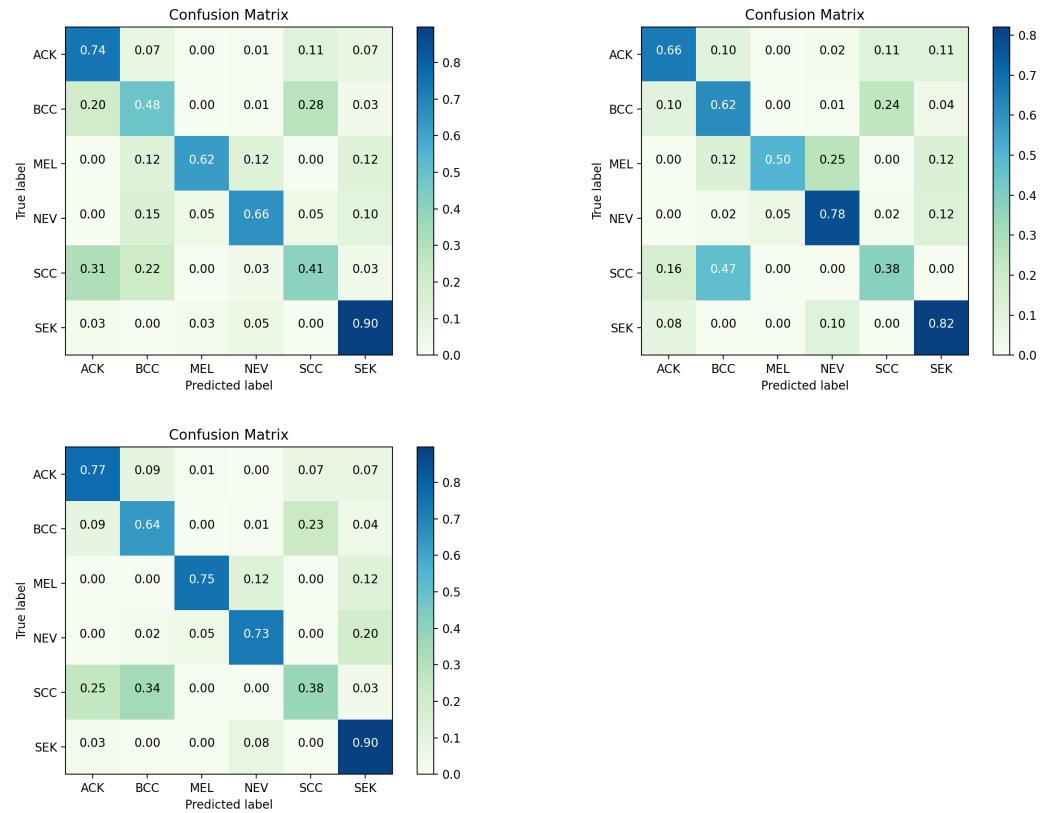


Figura 17 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o treinamento direto com *PAD-UFES-20 Dataset*

É possível visualizar que o resultado dessas matrizes de confusão são predominantemente positivo, com algumas classificações moderadas. Isso significa que o modelo obteve uma taxa geral de classificações corretas relativamente alta para a maioria das classes. No entanto, também podemos observar algumas classificações moderadas em algumas classes, como BCC e SCC, o que era esperado por conta da grande semelhança entre os dois tipos de câncer ([PACHECO; KROHLING, 2020](#)).

Com base nessas análises objetivas, podemos concluir que o modelo alcançou um desempenho razoável, embora haja variação nas métricas para diferentes classes.

4.4.3 Análise da transferência de aprendizado

Em seguida, com a base de comparação estabelecida, o pré-treinamento adquirido a partir do *Reddit Dataset* foi incorporado ao processo de treinamento da rede neural. Essa etapa permite explorar o potencial da transferência de aprendizado com dados fracamente rotulados e avaliar sua aplicabilidade para a tarefa em questão. As mesmas métricas previamente definidas serão novamente empregadas para avaliar o desempenho dessa nova configuração, a fim de realizar uma análise comparativa detalhada entre a abordagem com e sem transferência de aprendizado. Para ambas as abordagens, foram realizados o mesmo número de testes, utilizando as 3 arquiteturas mencionadas na Seção 4.4.

A seguir, podemos visualizar os resultados obtidos para o treinamento com *PAD-UFES-20 Dataset* após ser incorporado o pré-treinamento obtido com *Reddit Dataset*.

Rede	Loss	Balanced Accuracy	F1-Score	AUC
Resnet-50	1.215 ± 0.043	0.510 ± 0.028	0.536 ± 0.019	0.845 ± 0.009
Densenet-121	1.086 ± 0.036	0.570 ± 0.033	0.554 ± 0.031	0.870 ± 0.017
VGGnet-13	0.958 ± 0.062	0.621 ± 0.041	0.66 ± 0.025	0.905 ± 0.007

Tabela 5 – Resultado da transferência de aprendizado.

Ao compararmos as tabelas 4 e 5, referente ao treinamento direto com a base de imagens do *PAD-UFES-20 Dataset* e o processo de transferência de aprendizado, respectivamente, é possível observar algumas diferenças significativas. A perda média aumentou, indicando uma maior dificuldade do modelo em minimizar o erro durante o treinamento. Isso sugere que a incorporação do pré-treinamento no *Reddit Dataset* pode ter introduzido complexidades adicionais ou interferências no ajuste aos dados do *PAD-UFES-20 Dataset*.

A acurácia balanceada média também diminuiu, indicando que o modelo teve um desempenho inferior na classificação correta das classes, em comparação com o treinamento direto. Esse resultado pode ser atribuído à introdução de informações do *Reddit Dataset*, que pode conter padrões diferentes ou até mesmo fora do escopo, afetando a capacidade do modelo de generalizar para o segundo conjunto de dados.

O *F1-Score* médio também foi reduzido, indicando uma diminuição na harmonia entre a precisão e o *recall* das previsões. Isso também aponta para um possível introdução ruídos e informações conflitantes, prejudicando a capacidade do modelo de equilibrar a precisão e a cobertura nas previsões.

Em relação ao *AUC* médio, também é notada uma diminuição, indicando que o modelo teve um desempenho inferior na classificação entre as classes positivas e negativas em comparação com o treinamento direto. É perceptível que a incorporação do pré-treinamento pode ter afetado a capacidade do modelo de discriminar corretamente as classes e capturar as relações de ordem entre as previsões. As curvas *ROC* em questão podem ser observadas a seguir nas Figuras 21, e as matrizes na Figura 19.

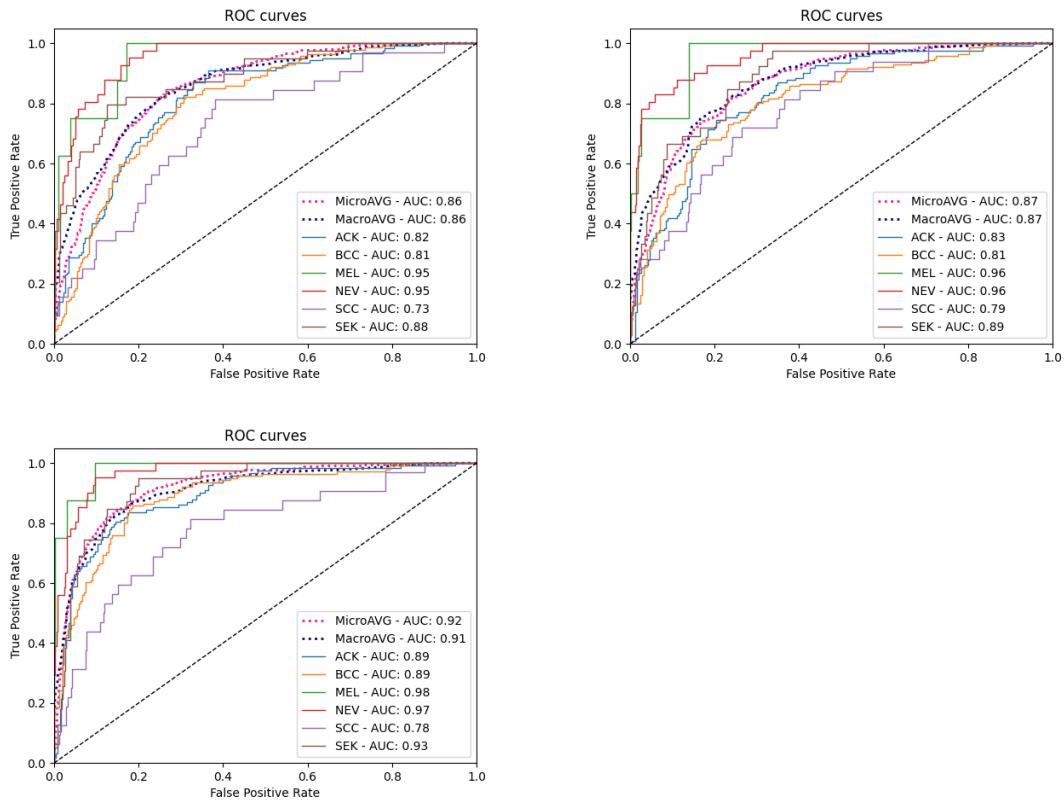


Figura 18 – Curvas *ROC* obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o *PAD-UFES-20 Dataset*

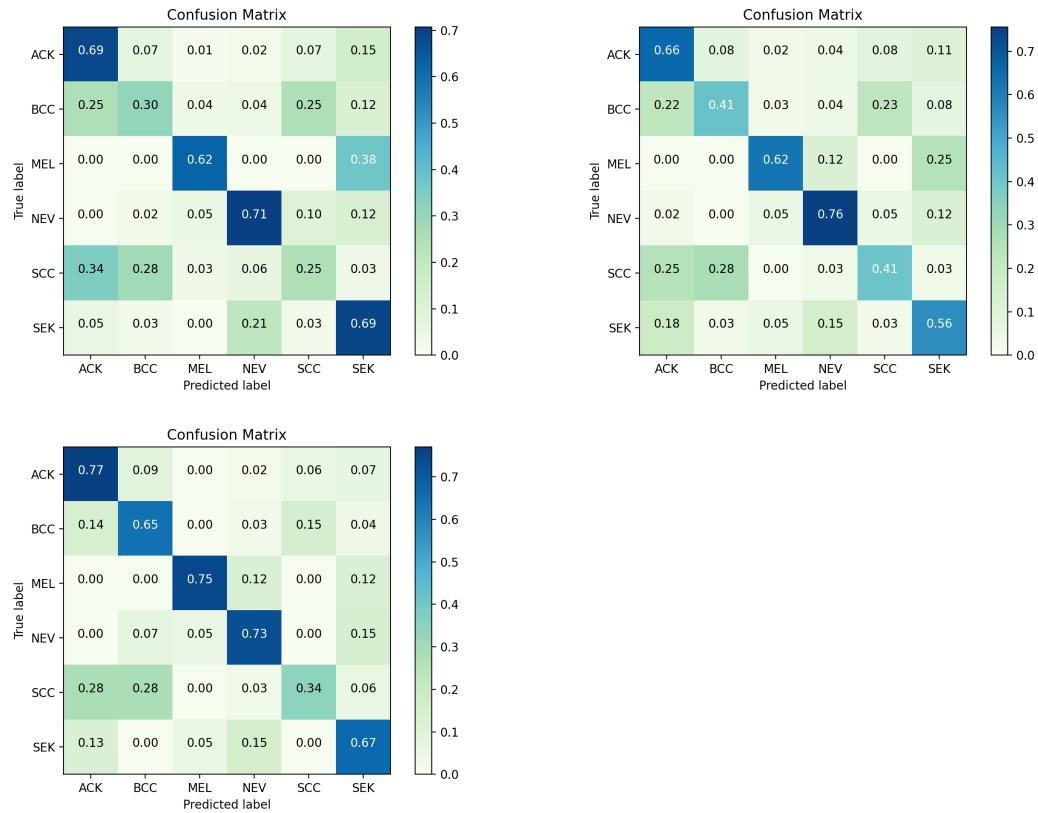


Figura 19 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o *PAD-UFES-20 Dataset*

Portanto, com base nessa análise comparativa, podemos concluir que a transferência de aprendizado, utilizando o pré-treinamento no *Reddit Dataset*, teve um impacto negativo no desempenho do modelo treinado com o *PAD-UFES-20 Dataset*. Isso indica que, para esse problema específico, o treinamento direto com o *PAD-UFES-20 Dataset* pode ser mais eficaz do que a transferência de aprendizado. É importante considerar as características dos conjuntos de dados, os padrões específicos do problema e a relevância do pré-treinamento ao decidir a melhor abordagem para treinar um modelo.

Também é importante ressaltar que os dados obtidos do Reddit, utilizados no pré-treinamento, apresentam certas limitações, estas que já foram citadas nas seções 3.1 e 4.1, que podem ter impactado os resultados da transferência de aprendizado. Essas limitações não são tão fáceis de se lidar, uma vez que a maioria é devida a qualidade dos dados que dependem única e exclusivamente do usuário que a submeteu, este que, geralmente, não possui entendimento da melhor forma de se registrar a imagem.

Esses problemas no Reddit Dataset podem ter levado a uma introdução de informações inconsistentes, conflitantes durante a transferência de aprendizado, resultando em uma diminuição no desempenho do modelo treinado originalmente com os dados do PAD-UFES-20. Além disso, é possível que as características específicas do *Reddit Dataset* não sejam totalmente relevantes ou adequadas para a tarefa de classificação do *PAD-UFES-20*.

Dataset.

Em resumo, os problemas citados anteriormente relacionados ao *Reddit Dataset* podem ter contribuído para a queda no desempenho do modelo treinado com a transferência de aprendizado. É importante realizar uma análise cuidadosa dos conjuntos de dados e considerar suas características antes de decidir sobre a incorporação do pré-treinamento em uma determinada tarefa de aprendizado de máquina.

4.4.4 Reddit x ImageNet

A seguir, é realizada a comparação entre os resultados do treinamento da rede neural utilizando o pré-treinamento com o *Reddit Dataset* e aqueles obtidos com o pré-treinamento utilizando dados da *ImageNet*. Então, realiza-se a análise para verificar se o treinamento com o *Reddit Dataset* proporcionou alguma vantagem em relação ao desempenho do modelo em comparação com o pré-treinamento na base *ImageNet*. Analisaremos as mesmas métricas utilizadas anteriormente para avaliar essa diferença de desempenho.

Para realizar essa comparação, um segundo treinamento da rede neural utilizando o *Reddit Dataset* foi realizado, dessa vez, sem a utilização dos pesos pré-treinados da *ImageNet*. Esse treinamento gerou um conjunto de pesos específicos para a base *Reddit*. Esses pesos são então utilizados na etapa de transferência de aprendizado durante o treinamento da rede neural com o *PAD-UFES-20*.

Por fim, é feita a comparação entre os resultados obtidos nas seções anteriores, onde se utilizaram pesos pré-treinados da *ImageNet*, e os resultados obtidos utilizando os pesos gerados a partir do treinamento com o *Reddit Dataset*. Para essa comparação, serão usadas as mesmas métricas anteriormente mencionadas, como a acurácia balanceada, o *F1-Score*, *AUC* e outras métricas relevantes.

Dessa forma, é possível avaliar se a transferência de aprendizado a partir do treinamento com o *Reddit Dataset* proporcionou alguma melhoria no desempenho do modelo em relação ao pré-treinamento na base *ImageNet*. Essa comparação fornece uma base para entender a eficácia relativa das duas estratégias de pré-treinamento, respondendo um dos questionamentos abordados neste estudo.

Em seguida, na tabela 6, tem-se os resultados da transferência de aprendizado, utilizando-se os pesos obtidos exclusivamente do treinamento com o *Reddit Dataset*.

Rede	Loss	Balanced Accuracy	F1-Score	AUC
Resnet-50	1.696 ± 0.073	0.297 ± 0.043	0.258 ± 0.041	0.622 ± 0.027
Densenet-121	1.682 ± 0.070	0.300 ± 0.036	0.304 ± 0.024	0.654 ± 0.041
VGGnet-13	1.424 ± 0.087	0.414 ± 0.049	0.396 ± 0.043	0.771 ± 0.019

Tabela 6 – Resultado da transferência de aprendizado com os pesos obtidos do treinamento com o *Reddit Dataset* sem influência da *imageNet*

Observando os resultados das tabelas 5 e 6, é possível notar que, de modo geral, o desempenho do modelo na primeira tabela (com pesos da *ImageNet*) é superior ao da segunda tabela (sem pesos da *ImageNet*). O modelo treinado com estes pesos obteve uma menor perda, uma maior acurácia balanceada, um maior *f1-score* e uma maior *AUC* em comparação ao modelo treinado apenas com o *Reddit Dataset*. As curvas *ROC* em questão e as matrizes de confusão obtidas com esses experimentos podem ser visualizadas nas Figuras 20.

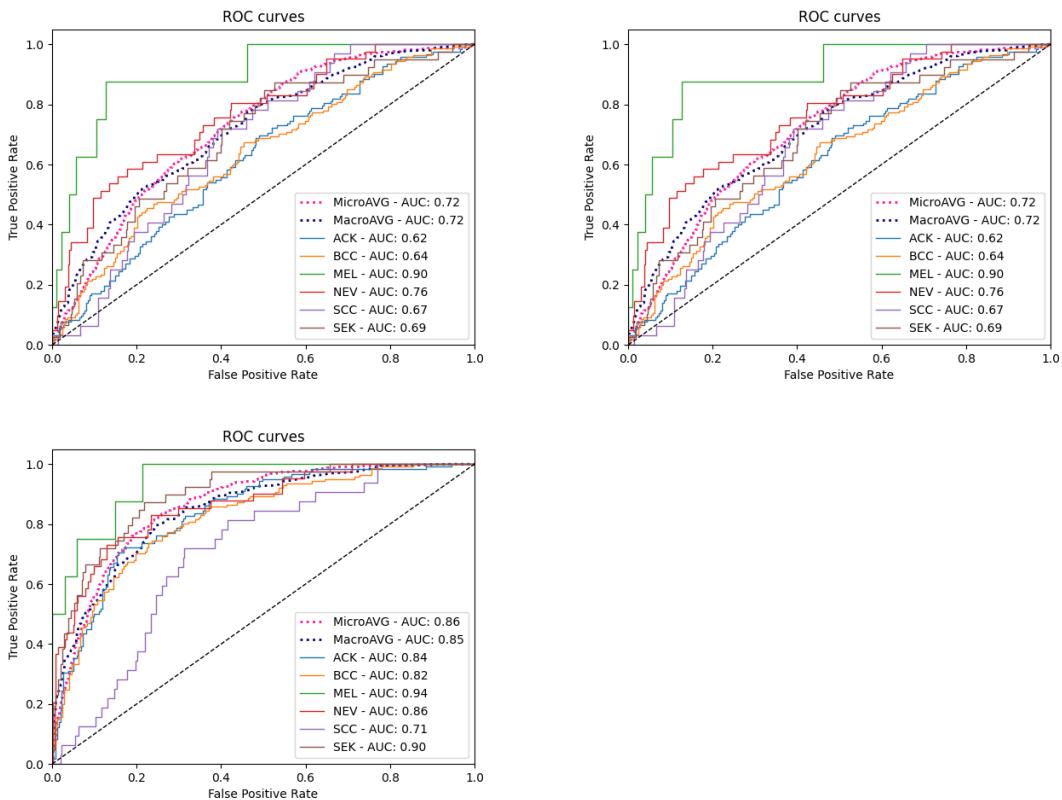


Figura 20 – Curvas *ROC* obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o *PAD-UFES-20 Dataset*, sem a utilização dos pesos da *ImageNet*.

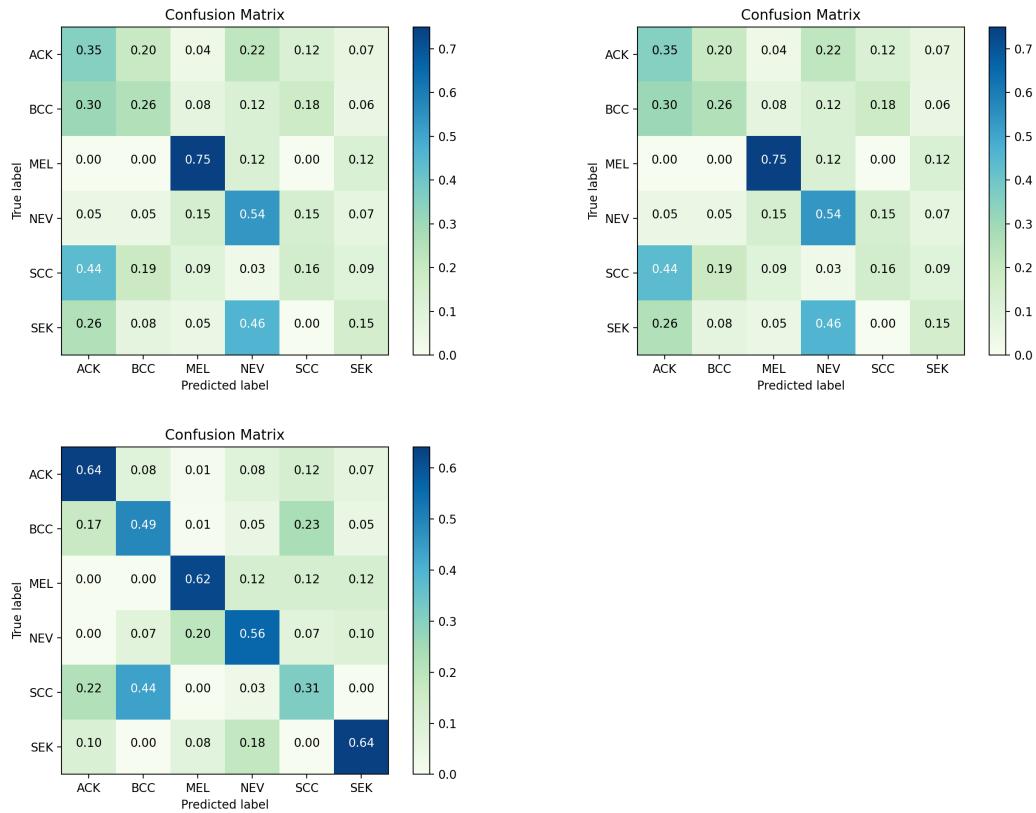


Figura 21 – Matrizes de confusão obtidas para os modelos Resnet, Densenet e VGGnet (em ordem), após o ajuste da rede com o *PAD-UFES-20 Dataset*, sem a utilização dos pesos da *ImageNet*.

Essa diferença de desempenho indica que a utilização dos dados fracamente rotulados do *Reddit Dataset*, sem a influência dos pesos da *ImageNet*, não foi capaz de fornecer um treinamento efetivo para o modelo. Na verdade, a incorporação dos dados do Reddit Dataset parece ter prejudicado o desempenho do modelo em comparação ao treinamento direto utilizando os pesos da *ImageNet*.

Essa análise sugere que os dados fracamente rotulados do *Reddit Dataset* não foram tão eficazes quanto os pesos pré-treinados da *ImageNet* para melhorar o desempenho do modelo. Portanto, podemos inferir que a diferença no desempenho final do modelo é mais atribuída à utilização dos pesos da *ImageNet* do que aos dados fracamente rotulados do *Reddit Dataset*.

Além disso, é importante ressaltar que o treinamento direto utilizando o *PAD-UFES-20 Dataset* e os pesos da *ImageNet* observado na tabela 4 apresentou um desempenho ainda melhor do que os resultados obtidos nas duas tabelas. Isso sugere que a utilização do *Reddit Dataset*, na verdade, prejudicou o treinamento e resultou em um desempenho inferior ao treinamento direto com os pesos da *ImageNet*.

Esses resultados indicam a importância de explorar abordagens alternativas para melhor utilizar os dados fracamente rotulados, em vez de simplesmente descartá-los.

Embora a incorporação direta dos dados do *Reddit Dataset* não tenha proporcionado o desempenho esperado, isso não significa que os dados fracamente rotulados sejam inúteis. Pelo contrário, existem outras metodologias e abordagens que podem ser exploradas para maximizar seu potencial.

5 Conclusão

Considerando os resultados obtidos e a discrepância na quantidade de dados entre o *Reddit Dataset*, com 18 mil imagens, e o *PAD-UFES-20 Dataset*, com apenas 3 mil imagens, podemos realizar uma análise dos fatores que podem ter influenciado o desempenho do modelo.

A utilização de dados com rótulos fracos do Reddit Dataset não resultou em uma melhoria do desempenho do modelo treinado com o *PAD-UFES-20 Dataset*. Pelo contrário, observou-se uma piora no desempenho. Isso indica que, nesse caso específico, a utilização dessa base dados com rótulos fracos não foi efetiva para suprir a falta de dados especializados. Os rótulos fracos presentes no Reddit Dataset podem ter introduzido ruído e imprecisão na fase de treinamento, afetando negativamente o desempenho final do modelo no *PAD-UFES-20 Dataset*.

Em relação à quantidade de dados com rótulos fracos, não foi possível formular uma resposta conclusiva sobre a sua influência no desempenho do modelo. Como o desempenho piorou com a incorporação dos dados fracos do Reddit Dataset, não podemos afirmar que uma maior quantidade desses dados teria impacto positivo. Pelo contrário, é possível que uma maior quantidade de dados com rótulos fracos tenha contribuído para uma maior presença de ruído no treinamento, resultando em uma piora no desempenho do modelo.

Além disso, é notável que o treinamento direto com o *PAD-UFES-20 Dataset* utilizando pesos da *ImageNet* apresentou um desempenho ainda melhor do que os resultados obtidos para os casos estudados de transferência de aprendizado. Isso sugere que a utilização do *Reddit Dataset*, na verdade, prejudicou o treinamento e resultou em um desempenho inferior ao treinamento direto com os pesos da *ImageNet*.

Diante desses resultados, é importante explorar novas metodologias e soluções que possam melhorar a utilização de dados fracamente rotulados, buscando aumentar a quantidade de dados disponíveis para treinamento. Uma possível abordagem é o uso de técnicas de aprendizado autossupervisionado, que podem permitir a geração de rótulos fracos de forma mais confiável e precisa. Essas técnicas envolvem a criação de tarefas auxiliares para pré-treinamento do modelo, sem depender de rótulos externos, e posterior ajuste fino com os dados especializados.

Além disso, é interessante investigar outras metodologias de aprendizado de máquina, como o aprendizado semi-supervisionado e o aprendizado ativo, que podem auxiliar na utilização eficiente de dados fracos. O aprendizado semi-supervisionado pode utilizar tanto dados rotulados quanto não rotulados para melhorar o desempenho do modelo, enquanto o aprendizado ativo visa selecionar de forma inteligente os exemplos a serem

rotulados, otimizando o uso dos recursos disponíveis.

Essas abordagens representam possíveis direções para futuras pesquisas e experimentos, para aprimorar a utilização de dados fracos e superar os desafios mencionados. Através do desenvolvimento e avaliação de novas metodologias, é possível buscar melhorias no desempenho dos modelos treinados com dados fracos, aumentando a sua capacidade de generalização e aplicabilidade em diferentes domínios.

Referências

- ACS. *American Cancer Society - Skin Cancer*. 2021. <<https://www.cancer.org/cancer/skin-cancer.html>>. Last accessed: 30 Jan 2023. Citado na página 9.
- BENGIO, Y. Learning deep architectures for ai. *Foundations*, v. 2, p. 1–55, 01 2009. Citado na página 10.
- BRASIL. *Lei N° 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais*. 2018. <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>. Lei Geral de Proteção de Dados Pessoais, Acesso em: 29 de junho de 2023. Citado na página 11.
- DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. *2009 IEEE conference on computer vision and pattern recognition*. [S.l.], 2009. p. 248–255. Citado na página 12.
- DILDAR, M. et al. Skin cancer detection: A review using deep learning techniques. *International Journal of Environmental Research and Public Health*, v. 18, n. 10, 2021. ISSN 1660-4601. Disponível em: <<https://www.mdpi.com/1660-4601/18/10/5479>>. Citado na página 10.
- DRAELOS, R. *Measuring Performance: AUC, AUROC*. 2019. <<https://glassboxmedicine.com/2019/02/23/measuring-performance-auc-auroc/>>. Acessado em: 01/07/2023. Citado 2 vezes nas páginas 4 e 30.
- FORNAZIN, M.; RACHID, R. R.; NETO, G. C. C. A saúde digital nos últimos quatro anos e os desafios para o novo governo. *Revista Eletrônica de Comunicação, Informação e Inovação em Saúde*, v. 16, n. 4, p. 753–758, dez. 2022. Disponível em: <<https://www.reciis.icict.fiocruz.br/index.php/reciis/article/view/3515>>. Citado na página 11.
- GERON, A. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media, 2017. ISBN 978-1491962299. Citado 9 vezes nas páginas 4, 13, 15, 17, 18, 19, 21, 29 e 31.
- HAYKIN, S. S. *Neural networks and learning machines*. Third. Upper Saddle River, NJ: Pearson Education, 2009. Citado na página 14.
- HE, K. et al. *Deep Residual Learning for Image Recognition*. 2016. 770-778 p. Citado na página 35.
- HUANG, G. et al. Densely connected convolutional networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2017. p. 2261–2269. Citado na página 35.
- INCA. *INSTITUTO NACIONAL DE CÂNCER. Estimativa 2023*. Rio de Janeiro, 2023. <<https://www.gov.br/inca/pt-br/assuntos/cancer/numeros/estimativa/>>. Acesso em: 9 de julho de 2023. Citado na página 9.

- JAMES, G. et al. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2013. Disponível em: <<https://faculty.marshall.usc.edu/gareth-james/ISL/>>. Citado 3 vezes nas páginas 22, 30 e 31.
- LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, p. 2278 – 2324, 12 1998. Citado 3 vezes nas páginas 4, 16 e 17.
- MASOOD, A.; AL-JUMAILY, A. Computer aided diagnostic support system for skin cancer: A review of techniques and algorithms. *International journal of biomedical imaging*, v. 2013, p. 323268, 01 2013. Citado na página 10.
- NAHATA, H.; SINGH, S. P. Deep learning solutions for skin cancer detection and diagnosis. In: _____. *Machine Learning with Health Care Perspective: Machine Learning and Healthcare*. Cham: Springer International Publishing, 2020. p. 159–182. ISBN 978-3-030-40850-3. Disponível em: <https://doi.org/10.1007/978-3-030-40850-3_8>. Citado na página 10.
- NALEPA, J.; MARCINKIEWICZ, M.; KAWULOK, M. Data augmentation for brain-tumor segmentation: A review. *Frontiers in Computational Neuroscience*, v. 13, p. 83, 12 2019. Citado 3 vezes nas páginas 4, 21 e 22.
- PACHECO, A. *Introdução a Redes Neurais Artificiais*. 2015. <<http://computacao-inteligente.com.br/algoritmos/conceitos/redes-neurais-artificiais>>. Acessado em: 10 de julho de 2023. Citado 2 vezes nas páginas 4 e 15.
- PACHECO, A. *Avaliação de modelos, cross-validation e data leakage*. 2021. <<http://computacao-inteligente.com.br/conceitos/avaliando-performance-cross-validation/>>. Acessado em: 12 de julho de 2023. Citado 2 vezes nas páginas 4 e 22.
- PACHECO, A. G.; KROHLING, R. A. The impact of patient clinical information on automated skin cancer detection. *Computers in Biology and Medicine*, v. 116, p. 103545, 2020. ISSN 0010-4825. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0010482519304019>>. Citado 2 vezes nas páginas 39 e 41.
- PACHECO, A. G. et al. Pad-ufes-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones. *Data in Brief*, v. 32, p. 106221, 2020. ISSN 2352-3409. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S235234092031115X>>. Citado na página 26.
- PACHECO, A. G. C. Agregação de classificadores neurais via integral de choquet com respeito a uma medida fuzzy. In: . [S.l.: s.n.], 2016. Citado 2 vezes nas páginas 4 e 14.
- PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, n. 10, p. 1345–1359, 2010. Citado na página 20.
- PASSOS, J. *Falta de integração e distribuição das bases de dados fragiliza sistemas de saúde*. 2022. <<https://www.epsjv.fiocruz.br/noticias/reportagem/falta-de-integracao-e-distribuicao-das-bases-de-dados-fragiliza-sistemas-de>>. Acesso em: 29 de junho de 2023. Atualizado em 1º de julho de 2022. Citado na página 11.
- PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Citado na página 28.

- PETRIE, T. et al. Melanoma early detection: Big data, bigger picture. *Journal of Investigative Dermatology*, v. 139, n. 1, p. 25–30, 2019. ISSN 0022-202X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0022202X18323546>>. Citado na página 9.
- RAGHU, M. et al. Transfusion: Understanding transfer learning for medical imaging. In: _____. Red Hook, NY, USA: Curran Associates Inc., 2019. Citado na página 21.
- RUDER, S. An overview of gradient descent optimization algorithms. 09 2016. Citado na página 35.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *Nature*, v. 323, p. 533–536, 1986. Citado na página 16.
- SHEELA, K.; DEEPA, S. N. Review on methods to fix number of hidden neurons in neural networks. *Mathematical Problems in Engineering*, v. 2013, 01 2013. Citado na página 15.
- SHERBURN, D. *Weak labeling*. 2021. <<https://www.thoughtworks.com/en-sg/insights/blog/data-science-and-analytics/weak-labeling>>. Acessado em 11/07/2023. Citado na página 14.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014. Citado na página 35.
- SUNG, H. et al. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, v. 71, n. 3, p. 209—249, May 2021. ISSN 0007-9235. Disponível em: <<https://doi.org/10.3322/caac.21660>>. Citado na página 9.
- TAKIDDIN, A. et al. Artificial intelligence for skin cancer detection: A scoping review (preprint). *Journal of Medical Internet Research*, v. 23, 07 2020. Citado na página 10.
- TURPAULT, N.; SERIZEL, R.; VINCENT, E. Limitations of weak labels for embedding and tagging. 02 2020. Citado na página 38.
- VIDYA, M.; KARKI, M. V. Skin cancer detection using machine learning techniques. In: *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. [S.l.: s.n.], 2020. p. 1–5. Citado na página 10.
- YOSINSKI, J. et al. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems (NIPS)*, v. 27, 11 2014. Citado na página 20.
- ZEILER, M. D.; FERGUS, R. Visualizing and understanding convolutional networks. In: FLEET, D. et al. (Ed.). *Computer Vision – ECCV 2014*. Cham: Springer International Publishing, 2014. p. 818–833. ISBN 978-3-319-10590-1. Citado na página 20.