

UNIVERSIDADE PRESBITERIANA MACKENZIE
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA ELÉTRICA E COMPUTAÇÃO

ANDRÉ PIRES DOS SANTOS

**ANÁLISE E COMPARAÇÃO DO DESEMPENHO DE DIFERENTES
ARQUITETURAS DE REDES NEURAIS ARTIFICIAIS PROFUNDAS
APLICADAS À TRIAGEM DE LESÕES DE PELE**

SÃO PAULO
2019

ANDRÉ PIRES DOS SANTOS

**ANÁLISE E COMPARAÇÃO DO DESEMPENHO DE DIFERENTES
ARQUITETURAS DE REDES NEURAIS ARTIFICIAIS PROFUNDAS
APLICADAS À TRIAGEM DE LESÕES DE PELE**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e Computação da Universidade Presbiteriana Mackenzie como parte dos requisitos para o título de Mestre em Engenharia Elétrica e Computação.

Orientador: Prof. Dr. Leandro Augusto da Silva

SÃO PAULO
2019

S231a Santos, André Pires dos

Análise e comparação do desempenho de diferentes arquiteturas de redes neurais artificiais profundas aplicadas à triagem de lesões de pele / André Pires dos Santos – São Paulo, 2019.

92f.: il., 30 cm.

Mestrado (Mestrado em Engenharia Elétrica e Computação) - Universidade Presbiteriana Mackenzie - São Paulo, 2019.

Orientador: Prof. Dr. Prof. Dr. Leandro Augusto da Silva

Bibliografia: f.

1. Redes neurais profundas 2. Visão computacional 3. Lesões de pele 4. Dermatologia I. Silva, Leandro Augusto da; orientador. II. Título.

CDD 006.13

ANDRÉ PIRES DOS SANTOS

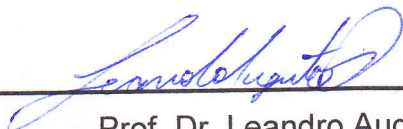
ANÁLISE E COMPARAÇÃO DO DESEMPENHO DE DIFERENTES
ARQUITETURAS DE REDES NEURAIS ARTIFICIAIS PROFUNDAS
APLICADAS À TRIAGEM DE LESÕES DE PELE

Dissertação de Mestrado apresentada
ao Programa de Pós-Graduação em
Engenharia Elétrica e Computação da
Universidade Presbiteriana Mackenzie,
como requisito parcial para a obtenção
do título de Mestre em Engenharia
Elétrica e Computação.

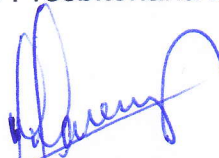
Orientador: Prof. Dr. Leandro Augusto
da Silva

Aprovado em 08 de agosto de 2019.

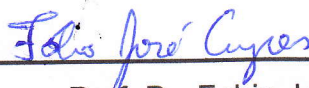
BANCA EXAMINADORA



Prof. Dr. Leandro Augusto da Silva
Universidade Presbiteriana Mackenzie



Prof. Dr. Mauricio Marengoni
Universidade Presbiteriana Mackenzie



Prof. Dr. Fabio Jose Ayres
Insper - Instituto de Ensino e Pesquisa

RESUMO

Este trabalho tem como proposta estudar o desempenho de diferentes arquiteturas de redes neurais profundas aplicadas na detecção de lesões dermatológicas. O resultado tem como aplicações secundárias a indicação do encaminhamento correto do paciente e prioridade de atendimento, aumentando assim a eficiência do fluxo de atendimento de pacientes carentes de acesso a essa especialidade médica. O estudo apresenta uma breve introdução dos principais motivos, a origem histórica, o impacto da falta de acesso da população ao sistema de saúde e os potenciais benefícios provenientes da inserção de inteligência artificial nos processos de atendimento. Apresenta ainda uma revisão de trabalhos científicos voltados à classificação de lesões de pele por algoritmos de inteligência artificial e os principais conceitos e histórico de redes neurais artificiais, desde estudos pioneiros até as arquiteturas mais recentes. Como resultados, o trabalho apresenta dois experimentos que somados contemplam 42 diferentes cenários de aplicação das redes neurais profundas para mensurar a acurácia dos algoritmos propostos em classificação de lesões de pele. Os experimentos avaliam o desempenho de diferentes arquiteturas, variando parâmetros e estratégias de inicialização de pesos das redes neurais. Como principais resultados, o trabalho apresenta que a arquitetura GoogLeNet treinada com 24.000 imagens, ao longo de mil épocas utilizando inicialização randômica dos pesos e taxa de aprendizado de 10^{-3} , foi capaz de obter uma acurácia de 89,72% para sugestão diagnóstica, 92,54% para indicação de prioridade e 96,03% para o encaminhamento apropriado do paciente em um conjunto de teste composto por 6.975 imagens.

Palavras-chave: redes neurais profundas, visão computacional, lesões de pele, dermatologia.

ABSTRACT

This work aims to study the performance of several deep neural network architectures applied to classify dermatological lesions. The result has as secondary applications to set the referral and priority of care to the patient, increasing the efficiency to deliver the appropriate treatment for patients lacking access to this medical specialty. The study presents a brief introduction of the main reasons, the historical origin and the impact from the lack of access of the population to the health system. The work also provides the potential benefits from the insertion of artificial intelligence in the health care processes. It also presents a review of scientific studies aimed at the classification of skin lesions by artificial intelligence algorithms and the main concepts and history of artificial neural networks from the beginning to the detailed description of the most recent architectures. As results, the work presents the results of two experiments that together contemplate 42 different scenarios of the application of deep neural networks to measure the accuracy of the proposed algorithms applied to classify skin lesions. The experiments evaluate the performance several architectures, varying parameters and weight initialization strategies. As a main result, the work presents that the GoogLeNet architecture trained with 24,000 images, during a thousand epochs using random initialization as strategy to initialize weights and learning rate of 10^{-3} , was able to obtain an accuracy of 89.72% for diagnostic suggestion, 92.54% for priority indication and 96.03% for patient referral in a 6,975-image test set.

Keywords: deep neural networks, computer vision, skin lesions, dermatology.

LISTA DE FIGURAS

Figura 1 – Imagem Microscopia de Epiluminação.....	12
Figura 2 – Imagem Microscopia Eletrônica de Transmissão.....	13
Figura 3 – Imagens Smatphone.....	14
Figura 4 – Anatomia de um neurônio biológico	20
Figura 5 – Arquitetura de um neurônio artificial	21
Figura 6 - Arquitetura Lenet-5 aplicada ao reconhecimento de dígitos	22
Figura 7 - Principais funções de ativação das redes neurais rasas	32
Figura 8 - Rede feedforward de camada única.....	38
Figura 9 - Rede feedforward de múltiplas camadas	39
Figura 10 - Rede recorrente	40
Figura 11 - Conexões esparsas	42
Figura 12 - Max Pooling	44
Figura 13 - Arquitetura AlexNet.....	48
Figura 14 - Arquitetura VGG	50
Figura 15 - Inception Module.....	52
Figura 16 - Aprendizado Residual.....	55
Figura 17 - Distribuição de lesões por grupo	60
Figura 18 - Imagens de exemplo das classes dos experimentos 1 e 2.....	61
Figura 19 - Imagens de exemplo das classes do experimento 3.....	64
Figura 20 - Curva de aprendizado do experimento 1	68
Figura 21 - Matrizes de confusão do experimento 1	69
Figura 22 - Curva de aprendizado do experimento 2	71
Figura 23 - Matrizes de confusão do experimento 2	72
Figura 24 - Curva de aprendizado do experimento 3 com TA 1-e03.....	74
Figura 25 - Curva de aprendizado do experimento 3 com TA 1-e06.....	75

Figura 26 - Curva de aprendizado do experimento 3 com TA 1-e09.....	75
Figura 27 - Matriz de confusão conjunto de teste - arquitetura GoogLeNet.....	78
Figura 28 - Matriz de confusão conjunto de teste - arquitetura VGG	79
Figura 29 - Matriz de confusão conjunto de teste - arquitetura ResNet	80
Figura 30 - Matriz de confusão definição de prioridade para atendimento	81
Figura 31 - Matriz de confusão de encaminhamentos	82

LISTA DE TABELAS

Tabela 1 – Resultados classificadores híbridos.....	10
Tabela 2 – Trabalhos Correlatos	11
Tabela 3 - Resultados AdaBoost e SVM.....	17
Tabela 4 - Comparação AdaBoost, SVM e Deep Learning	18
Tabela 5 - Disposição das camadas e parâmetros da arquitetura GoogLenet.....	53
Tabela 6 - Distribuição completa da arquitetura ResNet	54
Tabela 7 – Classes Alvo.....	58
Tabela 8 – Cenários Primeiro Experimento	61
Tabela 9 – Cenários Segundo Experimento	63
Tabela 10 – Cenários Terceiro Experimento	65
Tabela 11 - Resultados do experimento 1.....	67
Tabela 12 - Resultados do experimento 2.....	70
Tabela 13 - Resultados do experimento 3.....	73
Tabela 14 - Distribuição do Conjunto de Teste	76

SUMÁRIO

1	INTRODUÇÃO	1
1.1	Justificativa.....	3
1.2	Objetivos	4
1.3	Metodologia	5
1.4	Organização do documento	6
2	TRABALHOS CORRELATOS	7
3	REFERÊNCIAL TEÓRICO	12
3.1	Aquisição de imagens.....	12
3.2	Classificação de objetos em imagens	14
3.2.1	Classificadores Bayesianos.....	15
3.2.2	Classificadores Baseados em Instâncias	15
3.2.3	Máquinas de Vetores de Suporte (<i>SVM</i>).....	16
3.2.4	Ensemble	17
4	REDES NEURAIS ARTIFICIAIS.....	19
4.1	Conceitos Fundamentais de Redes Neurais Artificiais	19
4.2	Paradigmas de Aprendizagem em RNAs.....	23
4.2.1	Aprendizado Supervisionado.....	23
4.2.2	Aprendizado Não Supervisionado	24
4.3	Algoritmo de aprendizagem por correção de erro.....	26
4.3.1	Gradiente Descendente	27
4.3.2	Gradiente Descendente Estocástico	28
4.4	Transferência de Aprendizado.....	29
4.4.1	Inicialização Randômica (<i>IR</i>)	30
4.4.2	Extensão de Modelo com Reinicialização de Pesos (<i>EMRP</i>).....	30
4.4.3	Extensão de Modelo com Extratores Fixos (<i>EMEF</i>)	30
4.5	Funções de Ativação.....	30
4.5.1	Função de ativação Logística.....	32
4.5.2	Função de ativação Softmax	33
4.5.3	Rectified Linear Unit (ReLU).....	33
4.5.4	Leaky Rectified Linear Unit (LReLU)	35
4.5.5	Parametric Rectified Linear Units (PReLU).....	35
4.5.6	Exponential Linear Units (ELUs).....	36
4.5.7	Scaled Exponential Linear Unit (SELU)	36

4.6	Arquitetura de RNAs	37
4.6.1	Redes feedforward de camada simples.....	37
4.6.2	Redes feedforward de camada múltipla.....	38
4.6.3	Redes recorrentes.....	39
4.6.4	Redes Profundas.....	40
4.6.4.1	LeNet.....	45
4.6.4.2	AlexNet.....	47
4.6.4.3	VGGNet.....	49
4.6.4.4	GoogLeNet.....	50
4.6.4.5	Resnet.....	53
5	METODOLOGIA EXPERIMENTAL	56
5.1	Coleta de Dados.....	56
5.2	Rotulação das imagens e definição de classes alvo.....	57
5.3	Ferramentas utilizadas.....	58
5.4	Experimentos.....	59
5.4.1	Primeiro Experimento.....	59
5.4.2	Segundo Experimento.....	62
5.4.3	Terceiro Experimento.....	63
5.5	Apresentação dos Resultados.....	66
6	RESULTADOS EXPERIMENTAIS	67
6.1	Primeiro Experimento.....	67
6.2	Segundo Experimento.....	69
6.3	Terceiro Experimento.....	72
7	Conclusões e trabalhos futuros	83
8	REFERÊNCIAS BIBLIOGRÁFICAS	86
9	ANEXOS	95
9.1	Anexo 1 Agrupamento de CIDs.....	95

1 INTRODUÇÃO

O Acesso à saúde para a população é historicamente um problema crônico no Brasil. Para se ter como exemplo, na década de 60 a razão entre médicos e habitantes apresentava um percentual de apenas 0,0623%. Mesmo com um considerável aumento desta proporção ao longo dos anos, em 2018 o percentual é de aproximadamente 0,22% (SCHEFFER et al., 2018). Segundo os mesmos autores, este aumento de percentual ainda não reflete benefícios para a população.

Apesar de contar com 452.801 médicos em 2018, o Brasil ainda sofre com grande desigualdade na distribuição da população médica entre regiões, estados, capitais e municípios do interior (SCHEFFER et al., 2018). O percentual de distribuição de médicos pela população varia de 0,28% no sudeste para apenas 0,12% no norte. Somente o estado de São Paulo concentra 28% do total de médicos do país. Por sua vez, o Distrito Federal tem a razão mais alta, aproximadamente 0,44% dos habitantes, seguido pelo Rio de Janeiro, com 0,36%. Esta realidade se opõe drasticamente a de outras unidades federativas como o Maranhão e o Pará, onde essa distribuição de médicos por habitantes é representada por apenas 0,09% da população (SCHEFFER et al., 2018).

A situação se torna ainda mais complexa quando a assistência necessária requer um profissional especializado. Para ilustrar este cenário, os dados do estudo conduzido pela Sociedade Brasileira de Dermatologia indicam que em 2018 apenas 504 dos 5.565 municípios brasileiros possuem especialistas em dermatologia, ou seja, apenas 9,1%. (SCHMITT et al., 2014).

Segundo Vieira e colaboradores (2015), em estudo realizado no ano de 2014 em Minas Gerais, no município de Jequitinhonha, a especialidade Dermatologia foi a segunda mais acionada pelos médicos de atenção primária, totalizando 17,78% dos encaminhamentos e um tempo de espera médio superior a 108 dias (VIEIRA; LIMA; GAZZINELLI, 2015). A pesquisa realizada por Federman e colaboradores (1999), aponta para a variedade de lesões, tornando o diagnóstico desafiador para médicos de atenção primária e evidenciando, inclusive, o reflexo desta dificuldade na identificação de melanomas. Durante o estudo, os médicos dermatologistas foram capazes de identificar a morbidade em 93% de suas incidências contra apenas 52% dos médicos de outras especialidades, evidenciando assim a complexidade de diagnósticos desta natureza (FEDERMAN, 1999).

Esta distribuição heterogênea da população médica, especialmente em áreas remotas, contribui para o aumento da busca por meios disruptivos de atendimento como a

Telemedicina, a qual foi definida pela Organização Mundial de Saúde como um tipo de prestação de serviços de saúde à distância, por meio do uso de tecnologias de informação e comunicação (WORLD HEALTH ORGANIZATION, 2016). Nas últimas décadas, um número crescente de estudos tem mostrado que sua utilização é viável, eficaz, segura e pode ser custo-efetiva em diversas áreas da Medicina, e entre as mais estudadas e praticadas atualmente estão Teleradiologia, Telepatologia e Teledermatologia (WORLD HEALTH ORGANIZATION, 2016).

Os avanços tecnológicos das últimas décadas possibilitaram o desenvolvimento de dispositivos móveis como *tablets* e *smartphones* capazes de coletar e transmitir imagens, vídeos e dados em geral com grande qualidade e praticidade. Este avanço incentivou o desenvolvimento de novos modelos de atendimento como a Teledermatologia, onde imagens de lesões dermatológicas são capturadas e encaminhadas para a análise de um dermatologista por meios eletrônicos (PICCOLI et al., 2015). Este modelo de atendimento além de possibilitar que o paciente receba um diagnóstico com maior agilidade, evita a realização de consultas face a face e permite que médicos generalistas possam obter orientações de profissionais especializados para conduzir casos de maior complexidade (PICCOLI et al., 2015).

A efetividade, a acurácia e a confiabilidade dos diagnósticos dermatológicos via telemedicina foram amplamente estudados e são considerados equivalentes aos realizados por dermatologistas em consultas presenciais (LEVIN; WARSHAW, 2009). O estudo conduzido pelos autores por Levin e Warshaw (2009) registrou uma acurácia diagnóstica média de 77% para Teledermatologia e 72% obtidos em consultas dermatológicas convencionais. Os diagnósticos foram validados utilizando o exame anatomopatológico como o padrão-ouro.

A natureza digital destes atendimentos somada as recentes evoluções no campo de processamento de imagens proporcionaram um ambiente extremamente fértil para pesquisas relacionadas à aplicação de inteligência artificial na identificação e diagnóstico de lesões de pele. As pesquisas neste seguimento vêm demonstrando resultados promissores. Em 2016, Shrivastava e colaboradores desenvolveram um algoritmo baseado em *Support Vector Machine* (SVM) capaz de identificar Psoríase com 99,81% de precisão a partir de uma amostra de apenas 540 imagens previamente rotuladas (SHRIVASTAVA et al., 2016). Em 2018, na pesquisa realizada por Han e colaboradores, um algoritmo de classificação construído com redes neurais profundas registrou 91% de acurácia na tarefa de diferenciar lesões pigmentadas entre benignas e malignas utilizando uma amostra de 21.306 imagens (HAN et al., 2018).

Este trabalho visa ampliar os experimentos já realizados em Teledermatologia, observando a acurácia registrada por diferentes técnicas de inteligência artificial denominadas simplesmente por *Deep Learning*, identificando oportunidades de aumento da precisão dos algoritmos e avaliando os benefícios da inserção desse conceito no apoio a prática assistencial como uma possível ferramenta de suporte a decisão médica.

1.1 Justificativa

Na literatura há um grande volume de pesquisas apontando para uma satisfatória acurácia de algoritmos de inteligência artificial na identificação de diversos tipos de doenças de pele, desde as mais simples como onicomicose (HAN et al., 2018a) até as mais graves como melanomas (CODELLA et al., 2018). Entretanto, ao aprofundar na literatura e consultar especialistas no assunto sobre os resultados, conclui-se que os estudos de um modo geral foram focados em dizer se uma lesão pode ou não ser diagnosticada como uma determinada morbidade. Contudo, para aplicação das técnicas avaliadas em um âmbito de saúde populacional de grande escala, se faz necessário uma abordagem mais ampla, capaz de classificar a lesão diagnosticada em categorias estabelecidas e sugerir uma conduta adequada para o tratamento do paciente, qualquer que seja sua condição. Além do foco no diagnóstico de uma única doença, observa-se ainda que os estudos em geral foram realizados com protocolos e equipamentos de aquisição de imagens desenhados especificamente para estudos no âmbito de inteligência artificial. Ou seja, com o intuito de potencializar a acurácia dos algoritmos de classificação. Entretanto, estes protocolos de aquisição de imagens em geral consomem um tempo elevado de profissionais de saúde para executar a coleta e os equipamentos utilizados não se encontram disponíveis na maioria das unidades básicas de saúde. Estas características impossibilitam a aplicação dos algoritmos no atendimento da população no âmbito do sistema público de saúde. Portanto, apesar dos estudos anteriores nesta temática, identifica-se ainda uma oportunidade de avaliar o desempenho destas técnicas em um contexto mais amplo, que não se proponha apenas a classificar lesões como pertencentes ou não a um determinado diagnóstico, mas sim em uma classificação mais abrangente utilizando protocolos simples que possam ser aplicados por profissionais de atenção primária no dia a dia e equipamentos de aquisição de imagens disponíveis para as redes de atenção básica do Sistema Único de Saúde (SUS).

Justifica-se ainda que os trabalhos da literatura não foram conduzidos com foco em propor a utilização dos algoritmos em atendimentos do sistema público de saúde no sentido de tornar a operação mais eficiente a partir da aplicação de algoritmos inteligentes com a

finalidade de eliminar etapas desnecessárias do processo de diagnóstico, fornecendo aos profissionais de saúde uma ferramenta para apoio diagnóstico capaz de triar, sugerir tratamentos e direcionar rapidamente casos graves para especialistas, potencialmente modificando o desfecho do paciente em virtude da redução do tempo de espera pelo atendimento.

1.2 Objetivos

O objetivo geral deste trabalho é avaliar a viabilidade da aplicação de redes neurais artificiais profundas (*Deep Learning*) para triagem, apoio diagnóstico e sugestão de tratamentos, com base em protocolos de aquisição de imagens aplicáveis e equipamentos disponíveis nas redes de atenção primária do Sistema Único de Saúde em um contexto de atendimento a livre demanda populacional do sistema público de saúde.

O trabalho não propõe automatizar totalmente o processo de avaliação e diagnóstico, mas sim oferecer aos profissionais de saúde uma ferramenta de apoio para confirmar hipóteses diagnósticas, aumentar a produtividade na avaliação dos pacientes e reduzir o tempo de espera, encaminhando o paciente diretamente ao local adequado para o tratamento de sua lesão.

Como objetivos específicos da pesquisa, destacam-se os seguintes pontos:

- i. Avaliar a diferença de desempenho entre as arquiteturas de Deep Learning: ResNet, VGG e GoogLeNet. Aplicadas à tarefa descrita no objetivo principal.
- ii. Avaliar o impacto da transferência de aprendizado no desempenho das diferentes arquiteturas de redes neurais artificiais profundas avaliadas nos experimentos conforme a variação da estratégia de inicialização e ajuste dos pesos. Serão avaliadas as seguintes Estratégias: Inicialização Randômica, Fixação da extração de características e Extensão do modelo de predição.
- iii. Identificar a diferença da acurácia na classificação de lesões de pele obtida pelas redes neurais profundas quando utilizadas para classificar imagens dermatológicas coletadas por meio de protocolos de aquisição e equipamentos do estado da arte e pelas vias de atendimento e equipamentos disponíveis no Sistema Único de Saúde.

1.3 Metodologia

Para atingir os objetivos deste trabalho, o estudo consiste de um conjunto de dados particularmente heterogêneo, composto por 140.446 imagens correspondentes a 57.568 lesões dermatológicas. As imagens foram obtidas entre julho de 2017 e agosto de 2018 por meio de um projeto assistencial filantrópico fruto de uma parceria entre a Sociedade Beneficente Israelita Brasileira Albert Einstein e a prefeitura do município de São Paulo. Ao longo do projeto pacientes que aguardavam na fila para atendimento com dermatologistas no município foram direcionados a Unidades Básicas de Saúde onde fotos de suas lesões de pele foram coletadas por técnicas de enfermagem utilizando a câmera padrão do *smartphone* Samsung Galaxy S7 em consultórios convencionais. As lesões foram diagnosticadas por treze especialistas e classificadas em 210 CIDs (Código Internacional de Doenças) diferentes. O comitê de dermatologistas do Hospital Israelita Albert Einstein, proprietário desse conjunto de dados, construiu um protocolo para guiar os profissionais durante a avaliação de lesões de pele por meio da Tele dermatologia. O protocolo se baseia em um agrupamento de 210 CIDs em 17 categorias. A predição destas categorias possibilita, de acordo com o comitê, permite propor um tratamento ao paciente que possuir determinadas doenças mais simples, priorizar o atendimento de casos mais graves, identificar a necessidade ou não de coleta de biópsias ou mesmo o impedimento de uma avaliação à distância em virtude da necessidade de exames físicos.

1.4 Organização do documento

Além da introdução em que se apresenta o contexto da pesquisa e objetivo, o trabalho está estruturado como segue.

O Capítulo dois traz uma breve revisão de trabalhos relacionados à classificação de lesões de pele. Em seguida, o capítulo três introduz conceitos sobre as técnicas de inteligência artificial utilizadas nos trabalhos apresentados no capítulo dois.

O Capítulo quatro apresenta os conceitos das Redes Neurais Artificiais, o surgimento das redes neurais profundas e algumas de suas principais arquiteturas.

O capítulo cinco descreve a metodologia utilizada no projeto, desde a aquisição do conjunto de dados, a escolha das ferramentas utilizadas e os cenários de testes aplicados.

O capítulo seis apresenta os resultados obtidos. Por fim, o capítulo sete apresenta as conclusões e os trabalhos futuros propostos neste estudo.

2 TRABALHOS CORRELATOS

Pesquisas na área de inteligência artificial e reconhecimento de padrões em imagens aplicadas à medicina e mais especificamente a dermatologia estão progredindo rapidamente e apresentando resultados promissores que cominam para a adoção destas técnicas no dia a dia dos serviços de saúde.

A dificuldade para obtenção de dados, historicamente uma barreira para a condução de estudos desta natureza, esta sendo superada com o apoio de instituições como a *International Skin Imaging Collaboration (ISIC)* que agregou um amplo conjunto de dados de imagens dermatoscópicas e disponibilizou para acesso público. Atualmente, o conjunto de dados é composto por mais de 20.000 imagens capturadas em centros clínicos de diferentes países, por meio de uma variedade de dispositivos usados em cada centro. O conjunto de dados *ISIC* foi à base para o primeiro estudo comparativo público de diagnósticos por inteligência artificial, o desafio “*Challenge on Dermoscopic Image Analysis*” realizado em 2016 (GUTMAN et al., 2016).

Os objetos de investigação das pesquisas são amplos, e contemplam em seu escopo desde a avaliação da relevância de cada característica visual das lesões de pele para a classificação (OGORZALEK et al., 2010) até a acurácia do diagnóstico por algoritmos (ESTEVA et al., 2017).

Os autores Maglogiannis e Doukas em 2009 apresentaram uma revisão do estado da arte da classificação de lesões dermatológicas, abordando os principais padrões utilizados para aquisição de imagem, extração de características e classificação (MAGLOGIANNIS; DOUKAS, 2009).

Como principais métodos de aquisição de imagens são destacados: Microscopia de epiluminação (*ELM* ou dermatoscopia), Microscopia eletrônica de transmissão (*TEM*) e aquisição de imagem por meio de câmeras digitais.

Em relação à extração de características e classificação Maglogiannis e Doukas resgatam a regra ABCD, base para o diagnóstico realizado por dermatologistas que consiste na mensuração de quatro fatores, respectivamente: (A) Assimetria, (B) Bordas, (C) Cores e (D) Diferenciais de estrutura.

O estudo conduzido por Celebi e Mishra (2016), teve foco na detecção de melanoma, e destacou a importância da execução de técnicas de pré-processamento para obtenção de melhores resultados. As técnicas apresentadas incluem o tratamento da iluminação, definição de um canal de cores apropriado, conversão e normalização das cores no canal selecionado,

realce de contraste, recorte da imagem na região de interesse e remoção de objetos alheios ao objeto da classificação, como a presença de pelos por exemplo (MISHRA; CELEBI, 2016).

Celebi e Mishra (2016) mencionam como estratégia para ampliar a capacidade de generalização dos classificadores a aplicação de técnicas de pós-processamento após a extração de características. As principais técnicas apresentadas foram à fusão de regiões, suavização, remoção da península e realce das bordas da região de interesse.

As principais técnicas para classificação são Redes Neurais Artificiais (RNAs), Maquinas de vetores de suporte (SVM do inglês *Support Vector Machines*), Regressão Logística, Arvores de Decisão, K-nearest-neighbor (*kNN*), Classificadores bayesianos e algoritmos de aprendizagem profunda (MISHRA; CELEBI, 2016).

Pathan, Prabhu e Siddalingaswamy (2018), escreveram a revisão literária mais recente em que muitos dos conceitos apresentados por Maglogiannis e Doukas em 2009 e Celebi e Mishra em 2016 são reafirmados. O estudo destaca como as principais fases para construção de um algoritmo de classificação bem sucedido a coleta de imagens, a escolha apropriada das técnicas de segmentação, extração de características e algoritmos de classificação (MAGLOGIANNIS; DOUKAS, 2009) (MISHRA; CELEBI, 2016).

Na etapa de segmentação de imagens é salientando que este processo deve ser automatizado em ordem de se obter um processo de classificação mais apurado e uniforme. As principais técnicas de segmentação apresentadas são: Limiarização, Aumento da região da lesão, Reforço das bordas, Fusão, suavização e Deformação de modelos.

Em termos de extração de características as principais estratégias apresentadas são: Análise de padrões, Regra ABCD, Método Menzies, Lista de verificação de sete pontos, Algoritmo CASH, Características de forma, Características de cor, Características de textura e Características de alto relevo.

Após a análise das revisões literárias avaliadas para entender os estudos e abordagens aplicados na literatura em um contexto mais amplo, foram estudados mais detalhadamente trabalhos que apresentaram resultados de destaque para a tarefa de classificação de lesões de pele, com o intuito de nortear a escolha da abordagem a ser aplicada no problema alvo deste estudo.

Em 2016 Shrivastava e colaboradores desenvolveram um algoritmo para identificação de psoríase baseado em *Support Vector Machine* (SVM), obtendo índices de acurácia de aproximadamente 99,81% em um conjunto de dados composto por 540 imagens (SHRIVASTAVA et al., 2016).

Em razão de sua gravidade, a identificação de Melanoma foi alvo de um grande número de estudos. Mehta, Parihar e Mehta (2015) utilizaram uma rede neural de múltiplas camadas e obtiveram 96% de acurácia em um conjunto de dados composto por apenas 50 imagens (MEHTA; PARIHAR; MEHTA, 2015). Grzesiak-Kopec, Nowak e Ogorzałek (2015) desenvolveram um classificador utilizando os algoritmos *Naive Bayes*, *Random Forest* e *K* instance-based*, obtendo o desempenho de 92,2% de acurácia para um conjunto de dados de 1010 imagens (GRZESIAK-KOPEĆ; NOWAK; OGORZALEK, 2015). Os autores Oliveira, Pereira e Tavares (2017) utilizaram classificadores de conjuntos conhecidos como *Ensemble Classifiers* em um conjunto de dados de 1104 imagens e obtiveram um índice de aproximadamente 97% de acurácia (OLIVEIRA; PEREIRA; TAVARES, 2017).

Mesmo diante destes resultados, em razão da heterogeneidade e complexidade dos dados emergentes da dermatologia, as redes profundas vêm exercendo protagonismo nas tentativas de resolução de problemas de classificação desta natureza, sobretudo pelo fato das representações ou relacionamentos serem aprendidos por meio de dados brutos sem necessidade de pré-processamento (MIOTTO et al., 2017).

Várias arquiteturas de redes profundas foram aplicadas a detecção de lesões dermatológicas nos últimos anos. Han e colaboradores (2017) utilizaram o algoritmo *Microsoft ResNet-50* para classificação de imagens clínicas de 12 doenças de pele pigmentadas com o objetivo de diferenciá-las entre malignas e benignas. No trabalho usou-se um conjunto de dados com 20.826 imagens. O algoritmo treinado foi aplicado em dois conjuntos de teste: *Asan Test Dataset* e *Edinburgh Dataset*, obtendo a acurácia de 91% e 89% respectivamente. Os mesmos autores em um trabalho realizado no ano de 2018 apresentaram outra rede neural profunda baseada em detecção de região de interesse, no qual o algoritmo é treinado para distinguir a região da unha do background da imagem, resultando em um modelo híbrido de três arquiteturas (ResNet-50, VGG-19 e redes neurais *feedforward* de múltiplas camadas). Os resultados dos testes apresentaram valores de sensibilidade, especificidade e curva AUC de (96%, 95% e 98%), (83%, 97% e 95%), (92%, 79% e 93%) e (88%, 69% e 82%) para quatro conjuntos de dados completamente distintos entre si. O modelo foi treinado com um total de 49567 imagens para classificar onicomicose e foram utilizadas 1358 imagens para teste (HAN et al., 2018).

Abordagens híbridas englobando mais de um tipo de arquitetura também foram aplicadas a problemas de classificação de lesões de pele. No trabalho de Mahbod e colaboradores (2017) foram utilizadas 2000 imagens pertencentes às doenças: Melanoma,

Ceratose e Nevos Benignos. As imagens foram submetidas a cinco algoritmos de classificação (MAHBOD; ECKER; ELLINGER, 2017).

Os dois primeiros algoritmos testados utilizaram a arquitetura da rede neural profunda *AlexNet* pré-treinada pelo conjunto de dados *ImageNet* para realizar a extração de características, combinado com um algoritmo de *Support Vector Machines* para executar a classificação. Entretanto, no primeiro cenário as características foram extraídas da camada totalmente conectada denominada FC8 e no segundo cenário utilizou a saída da última camada totalmente conectada.

Os cenários três e quatro seguiram a mesma estratégia dos dois primeiros. Porém, a arquitetura de rede neural aplicada na extração de características foi a *VGG-16*.

O último cenário utilizou para treinamento do algoritmo de classificação uma fusão das saídas de ambas as arquiteturas aplicadas à extração de características.

A acurácia apresentada em todos os cenários avaliados pode ser observada na Tabela 1 onde Melanoma é representado como M e Ceratose como C (MAHBOD; ECKER; ELLINGER, 2017):

Tabela 1 – Resultados classificadores híbridos

Algoritmo	Camada de Extração	M-ACC	C-ACC	M-AUC	C-AUC	Média-AUC
AlexNet	FC8	83.3%	88%	78.4%	90.3	84.3%
AlexNet	Todas Camadas FC	84%	89.3%	82.8%	94.9%	88.9%
VGG-16	FC8	82.7%	79.3%	78.8%	81%	79.9%
VGG-16	Todas Camadas FC	85.3%	88%	81.9%	92.8%	87.3%
Fusão	Todas Camadas FC	84.7%	87.4%	84.8%	93.6%	89.2%

Fonte: MAHBOD; ECKER; ELLINGER, 2017

Bhattacharya e colaboradores (2017) também usaram uma abordagem utilizando arquitetura *AlexNet* combinado com *SVM*. O conjunto de dados era composto por 275 imagens para classificação de melanoma e obteve a acurácia aproximada de 80% (BHATTACHARYA et al., 2017).

Os trabalhos correlatos apresentados estão sintetizados na Tabela 2.

Tabela 2 – Trabalhos Correlatos

Autor e ano	Modelo	Imagens	Doença	Acurácia
Shrivastava, Vimal K., et al. (2015)	Support Vector Machine (SVM)	540	Psoríase	ACC - 99.81%
Mehta, A., Parihar, A. S., & Mehta, N. (2015)	Extreme learning machine (ELM) and proposed multi-layer feed- forward neural network	50	Melanoma	ACC - 96%
Grzesiak-Kopeć, K., Nowak, L., & Ogorzałek, M. (2015)	Naive Bayes classifier, Random Forest, K* instance-based classifier	1010	Melanoma	ACC – 92%
Codella, Noel C. F., et al. (2017)	Machine Learning Fusion Algorithm (16 algoritmos)	1.279	Melanoma	ACC 86%
Mahbod, A., Ecker, R., & Ellinger, I. (2017)	Deep Learning (AlexNet e VGG-16), SVM	2.000	Melanoma, Ceratose e Nevos	AUC - 89.2%
Zhang et al. (2017)	Deep Learning (GoogleNet Inception v3)	1595	Melanocytic Nevus, Ceratose Seborreica, Carcinoma Basocelular e Psoríase	Dataset A ACC - 86.54% Dataset B ACC - 85.86%
Bhattacharya et al. (2017)	Deep Learning (BVLC AlexNet) e SVM	275	Melanoma	ACC – 83%
Lopez et al. (2017)	Deep Learning (VGGNet)	842	Melanoma	ACC - 83.33%
Oliveira, R. B., Pereira, A. S., & Tavares, J. M. R. (2017)	Classificadores de Conjuntos (Ensemble)	1104	Melanoma	ACC - 94.3%
Ma, Z. & Tavares. J. M. R. (2017)	Support Vector Machine (SVM)	704	Melanoma	ACC - 90.5%
Vasconcelos, C. N., & Vasconcelos, B. N. (2017)	Deep Learning (GoogLeNet v1)	900	Melanoma	ACC - 0.83
Han et al (2018)	Deep Learning (ResNet-152 e VGG19)	50925	Onicomiose	ACC - 96.0%
Han et al (2018)	Deep Learning (Microsoft ResNet-152 model)	21.306	Carcinoma, Ceratose, Melanoma, Nevo, Melanocítico, Granuloma, Hemangioma, Dermatofibroma, Verruga	Asan: ACC – 91% Edinburgh: ACC – 89%

Fonte: Criada pelo autor

3 REFERÊNCIAL TEÓRICO

Conforme exposto durante a introdução, este trabalho tem como objetivo desenvolver um mecanismo de triagem para lesões de pele utilizando inteligência artificial. Atualmente, há um considerável entusiasmo relacionado a esta possibilidade de triagem para identificação de quadros dermatológicos (PATHAN; PRABHU; SIDDALINGASWAMY, 2018).

Para potencializar a acurácia dos algoritmos Garnavi e colaboradores (2012), defendem a divisão do processo de classificação em quatro fases: Aquisição de imagens, Segmentação, Extração de características e Algoritmos de classificação (GARNAVI et al., 2012). Este capítulo descrevera as principais técnicas aplicadas na literatura em cada fase.

3.1 Aquisição de imagens

A técnica conhecida como Microscopia de Epiluminação (*ELM*, do inglês *Epiluminescence Microscopy*) é capaz de fornecer uma inspeção mais detalhada da superfície das lesões pigmentadas da pele e torna a epiderme translúcida, ampliando a visibilidade de muitas características dérmicas. Este método conta também com uma técnica conhecida como Transiluminação que consiste em transmitir luz em formato de um anel da periferia para o centro da lesão por um ângulo de 45° formando um feixe de luz em um ponto focal a cerca de um centímetro da superfície da pele. Esta abordagem proporciona visualizar com sensibilidade apurada o fluxo sanguíneo, a vascularização e a pigmentação abaixo da superfície de um nevo (ZOURIDAKIS; DOSHI; MULLANI, 2004). Um exemplo de imagem coletada por meio desta técnica pode ser observada na Figura 1.

Figura 1 – Imagem Microscopia de Epiluminação

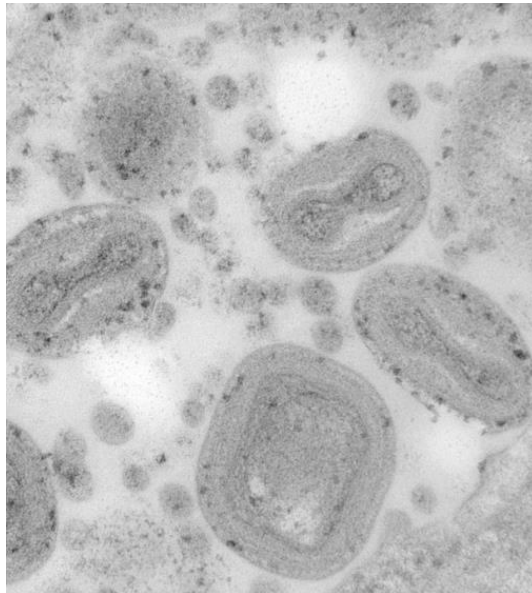


Fonte: ZOURIDAKIS; DOSHI; MULLANI, 2004.

A técnica denominada Microscopia Eletrônica de Transmissão ou *TEM* (do inglês *Transmission Electron Microscopy*) pode revelar a estrutura típica da organização de redes

elásticas na derme e, portanto, é usado principalmente para estudar o crescimento e a inibição do melanoma através de lipossomas (PATWARDHAN; DAI; DHAWAN, 2005). Um exemplo de imagem coletada por esta técnica pode ser observada na Figura 2.

Figura 2 – Imagem Microscopia Eletrônica de Transmissão



Fonte: PATWARDHAN; DAI; DHAWAN, 2005.

O uso de câmeras fotográficas digitais convencionais também é bastante comum em sistemas de inspeção de lesões cutâneas, particularmente para fins de telemedicina (LOANE et al., 1997). Entretanto, a qualidade da resolução em lesões muito pequenas com diâmetro inferior a meio centímetro frequentemente não atende aos requisitos desejados (MAGLOGIANNIS; DOUKAS, 2009). Algumas das imagens coletadas por esta técnica ao longo do projeto realizado entre o Hospital Albert Einstein e o Município de São Paulo podem ser observadas na Figura 3. Em razão da natureza da coleta, alguns ruídos foram identificados nas imagens, tais como a variação das condições de iluminação, a presença de objetos alheios à lesão nas imagens, à variação de distância entre o dispositivo e a lesão em diferentes coletas, a ausência de padronização de plano de fundo e o mau enquadramento da câmera no momento da coleta e em alguns casos, perda de qualidade da imagem devido ao movimento da câmera no momento da coleta.

Figura 3 – Imagens Smartphone

Fonte: Criada pelo autor

Outras técnicas menos comuns são utilizadas em contextos mais específicos para avaliação de melanomas, tais como: Tomografia Computadorizada (*TC*) que permite detectar melanomas e acompanhar o progresso da doença e a resposta ao tratamento. A tomografia por emissão de pósitrons (*PET* do inglês *Positron Emission Tomography*) também tem provado ser um método diagnóstico altamente sensível e adequado para detecção de várias neoplasias, incluindo o melanoma, complementando a imagem morfológica. A ressonância magnética também pode ser usada para delineamento de tumores. Tais métodos são utilizados principalmente para examinar o potencial metastático de um melanoma da pele e para posterior avaliação (PLEISS et al., 2007)

3.2 Classificação de objetos em imagens

O conceito de aprendizagem supervisionada por onde o algoritmo rotula objetos com base em um conjunto de exemplos previamente fornecidos durante uma etapa de treinamento (HAYKIN, 2008) é a abordagem mais utilizada para a construção de algoritmos de classificação de imagens dermatológicas, conforme o levantamento apresentado no capítulo 2.

Existem diversos tipos de classificadores consolidados na literatura, este subcapítulo apresenta um breve resumo dos classificadores aplicados nos trabalhos correlatos apresentados.

3.2.1 Classificadores Bayesianos

Os Classificadores Bayesianos foram apresentados originalmente por Duda e Hart (1973) com o algoritmo denominado *Nayve Bayes (NB)* (DUDA; HART, 1973). O algoritmo é conhecido por sua simplicidade, sobretudo quando comparado as Redes Neurais Profundas ou as Máquinas Vetores de Suporte (RISH, 2001). A simplicidade desta técnica advém da suposição realizada pelo algoritmo de que todos os atributos do conjunto de dados são independentes um do outro. Esta suposição é aferida no algoritmo pela forma como a classificação é realizada. Considerando um conjunto de dados de treinamento em que cada elemento é representado por um vetor de atributos x e cada atributo é denominado a_i onde i representa um atributo qualquer, os vetores são classificados como pertencentes a classes presentes em um vetor de y onde cada classe é denominada por y . O algoritmo calcula o percentual dos valores de cada atributo a_i do conjunto de dados que pertence a cada classe y . A classificação é então realizada pelo algoritmo por meio da probabilidade condicional pc obtida pela soma ponderada dos percentuais que representam a probabilidade de cada um dos atributos de x pertencerem a cada classe y . A classe com a maior probabilidade condicional será correspondente ao rótulo atribuído a x . Embora a suposição descrita não seja verdadeira na maioria das tarefas do mundo real, o algoritmo é capaz de executar classificações com acurácia elevada (FRIEDMAN; GEIGER; GOLDSZMIDT, 1997).

No contexto de classificação de lesões de pele, o estudo realizado por Grzesiak-Kopeć, Nowak, e Ogorzalek (2015) aplica está técnica para identificação de melanomas em lesões dermatológicas com resultado de 91,7% de acurácia, demonstrando a eficácia do algoritmo em aplicações práticas (GRZESIAK-KOPEĆ; NOWAK; OGORZALEK, 2015).

Rish destaca também bons resultados obtidos em estudos voltados para classificação de texto e predição do consumo de recursos computacionais (RISH, 2001).

3.2.2 Classificadores Baseados em Instâncias

Os classificadores baseados em instâncias mapeiam os atributos dos elementos do conjunto de dados de treinamento em hiperplanos multidimensionais. Para realizar a

classificação o algoritmo posiciona o elemento a ser classificado no hiperplano e rotula como pertencente à mesma classe dos elementos do treinamento com maior proximidade estatística, ou seja, a mesma classe dos vizinhos mais próximos (FIX; HODGES, 1951).

Um exemplo de algoritmo baseado em instancias é o *K-Nearest Neighbors*. O K-NN foi inicialmente apresentado por Fix e Hodges em 1951 (FIX; HODGES, 1951). Neste algoritmo *K* designa quantos vizinhos mais próximos devem ser considerados para definir a classe do objeto que se deseja classificar. Após a seleção dos *K* vizinhos pela proximidade estatística, o rotulo atribuído será determinado pela classe com maior número de vizinhos próximos selecionados.

O calculo da distância entre os vizinhos é obtido por meio da Distância Euclidiana (FIX; HODGES, 1951).

No estudo conduzido por Grzesiak-Kopeć, Nowak, e Ogorzalek em 2015 onde a técnica é aplicada na classificação de melanomas, o algoritmo alcançou 92,2% de acurácia (GRZESIAK-KOPEĆ; NOWAK; OGORZALEK, 2015).

3.2.3 Máquinas de Vetores de Suporte (SVM)

Apresentado pela primeira vez em 1992 por Vapnik (VAPNIK, 1998). O SVM é um algoritmo de aprendizado supervisionado onde os vetores de características são mapeados de forma não linear para um espaço de elementos multidimensional. Neste espaço multidimensional o algoritmo amplia a margem que denota as diferenças encontradas entre os exemplos que compõe cada classe, reduzindo a média da taxa de erro nos elementos do conjunto de treinamento e aumentando a margem geométrica dos objetos (VAPNIK, 1998) (Law, 2006).

SVMs utilizam funções de custo convexas sem mínimos locais para classificação (VAPNIK, 1998). No contexto de classificação de lesões dermatológicas, o classificador desenvolvido por Shrivastava e colaboradores obteve 99,81% de acurácia na tarefa de identificar Psoríase com base em um conjunto de dados composto por 540 imagens (SHRIVASTAVA et al., 2016). Ma e Tavares (2017) utilizaram um conjunto de dados de 704 imagens e registraram uma precisão de aproximadamente 91% na identificação de melanoma (MA; TAVARES, 2017). Em 2017 Bhattacharya e colaboradores construíram um algoritmo que utilizou a estrutura da Rede Neural AlexNet para extrair características substituindo as camadas de classificação por Maquinas de Vetores de Suporte e obteve uma acurácia de 83% (BHATTACHARYA et al., 2017).

3.2.4 Ensemble

As abordagens dos classificadores *Ensemble* combinam algoritmos de classificação para alcançar resultados mais precisos em comparação às abordagens de um único classificador (FREUND; SCHAPIRE, 1997). Vários métodos de *Ensemble* foram usados para a classificação de lesões de pele. O AdaptiveBoosting também conhecido como *AdaBoost* desenvolvido por Freund e Schapire (1997) é uma das variantes desta técnica de aprendizado. Tradicionalmente baseada em reforço, a ideia principal é adicionar classificadores fracos até que um erro mínimo seja alcançado (FREUND; SCHAPIRE, 1997).

AdaBoost concentra-se principalmente nas instancias difíceis de classificar. Se uma amostra não for classificada corretamente pelo classificador inicial do componente, ela será passada para a camada subsequente de classificação. Cada uma das áreas de amostras de treinamento possui pesos, o peso determina a probabilidade de uma amostra de treinamento ser selecionada para classificação (FREUND; SCHAPIRE, 1997).

No contexto de classificação de lesões de pele, pode-se citar como trabalhos de destaque o estudo realizado em 2011 por Capdehourat e colaboradores onde um algoritmo *AdaBoost* aplicado a classificação de lesões pigmentadas entre os rótulos maligna ou benigna superou a precisão das máquinas de vetores de suporte *SVM* tanto em cenários de segmentação manual como automatizada (CAPDEHOURAT et al., 2011). Os resultados são apresentados na tabela 3:

Tabela 3 – Resultados AdaBoost e SVM

Segmentação da Imagem	Classificador	Acurácia
Automática	AdaBoost	92%
Automática	SVM	89%
Manual	AdaBoost	93%
Manual	SVM	89%

Fonte: CAPDEHOURAT et al., 2011

Em 2016 Premaladha e Ravichandran realizaram um estudo propondo um classificador *AdaBoost* híbrido considerando o *SVM* como um classificador base que é treinado usando *AdaBoost* suave. A abordagem híbrida indicou melhores resultados em comparação com a abordagem de *AdaBoost* suave autônoma ou o *SVM* atuando de forma

independente (PREMALADHA; RAVICHANDRAN, 2016). O algoritmo proposto apresentou um desempenho inferior apenas quando comparado a uma rede neural profunda com arquitetura proposta pelos autores, construída com 15 camadas de extração de características e 2 camadas de classificação. Os resultados obtidos podem ser observados na Tabela 4 que apresenta a acurácia, a sensibilidade e a especificidade do algoritmo, esses indicadores de desempenho se referem ao percentual de classificações corretas do algoritmo, ao percentual de acerto entre os verdadeiros positivos e verdadeiros negativos respectivamente.

Tabela 4 – Comparação AdaBoost, SVM e Deep Learning

Classificador	Acurácia	Sensibilidade	Especificidade
Rede Neural Artificial	90%	93%	87%
SVM	90%	93%	87%
Rede Neural Profunda	93%	95%	90%
Adaboost Convencional	87%	90%	83%
AdaBoost Híbrido	92%	94%	89%

Fonte: PREMALADHA; RAVICHANDRAN, 2016

Mais recentemente em 2017, outro estudo da aplicação de classificadores de conjunto para identificação de melanomas foi conduzido pelos pesquisadores Oliveira, Pereira, e Tavares. O classificador por meio de um conjunto de dados de 1104 imagens obteve 94.3% de acurácia, 91.8% de sensibilidade e 96.7% de especificidade (OLIVEIRA; PEREIRA; TAVARES, 2017).

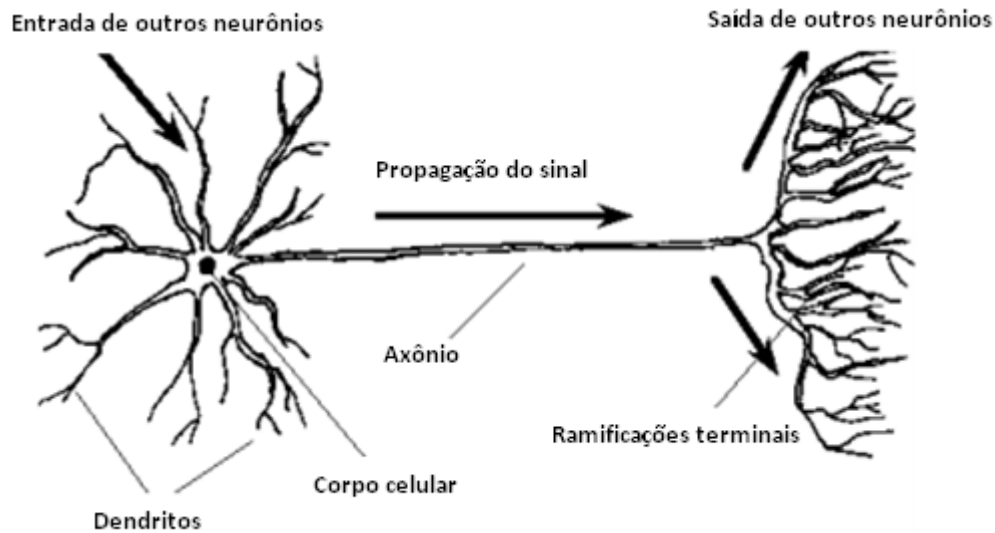
4 REDES NEURAIS ARTIFICIAIS

4.1 Conceitos Fundamentais de Redes Neurais Artificiais

Em 1943, McCulloch e Pitts criaram o primeiro modelo computacional de um neurônio artificial simplificado (MCCULLOCH; PITTS, 1943). Esse modelo analógico procurava representar matematicamente o comportamento de um neurônio biológico. Com os avanços das pesquisas nesta modelagem computacional do neurônio, a aplicação passou a ser empregada em diversos fins, em virtude de sua grande capacidade de aprendizado e generalização (HAYKIN, 2008). Caracterizadas como elementos capazes de processamento paralelo e solução de problemas não linearmente separáveis, as RNAs podem resolver uma grande variedade de problemas, como por exemplo, avaliação de imagens, reconhecimento facial, classificação de texto e voz, previsão de ações no mercado financeiro, identificação de anomalias em imagens médicas e muitas outras (HAYKIN, 2008; BASHEER; HAJMEER, 2000).

Inseridas na área de pesquisa conhecida como Inteligência Artificial, mais precisamente na abordagem conexionista (JAIN; MAO, 1996), as RNAs podem ser generalizadas por modelos matemáticos que processam as informações em vários elementos chamados neurônios. A analogia entre o comportamento do neurônio biológico, que tem seu modelo simplificado na Figura 4, e o seu processamento de informação possibilitou as pesquisas que deram base à modelagem de neurônios artificiais. Nesta figura, nota-se que os sinais chegam pelos dendritos e são propagados a outros neurônios pelo axônio. Portanto, a transmissão de sinal de um elemento a outro acontece por meio de conexões. Embora não esteja representado nesta figura, as conexões possuem um peso associado para ponderar o sinal transmitido. Cada neurônio (ou unidade) aplica uma função de ativação ao sinal transmitido, que em geral é não linear, determinando uma saída no neurônio (SILVA; ZUBEN, 1998). Os ajustes realizados nas conexões possibilitam a tomada de decisões baseadas em aprendizado.

Figura 4 – Anatomia de neurônio biológico

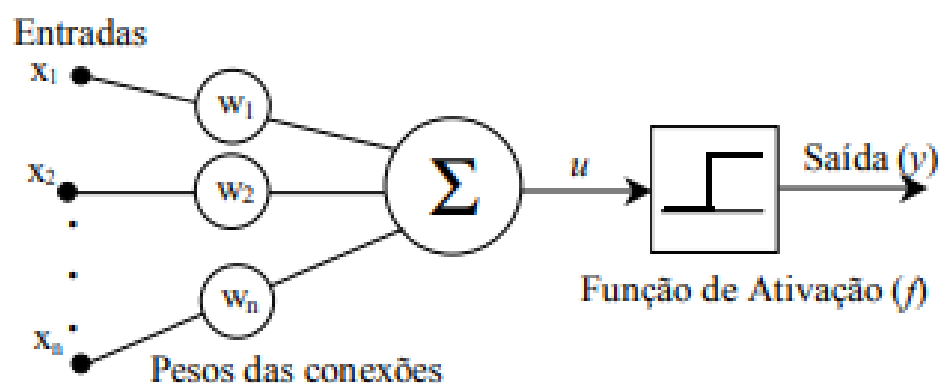


Fonte: SILVA; ZUBEN, 1998.

Mapeando as funcionalidades do neurônio biológico para o artificial, McCulloch e Pitts (1943) foram os primeiros pesquisadores a proporem essa modelagem para um neurônio, cuja arquitetura encontra-se ilustrada na Figura 5 (MCCULLOCH; PITTS, 1943). Nesta figura, os vetores de características transmitidos como sinais de entrada são representados por um conjunto de atributos $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$, sendo n o número total de entradas e, portanto, equivalente aos impulsos elétricos externos captados pelos dendritos (MCCULLOCH; PITTS, 1943).

As ponderações exercidas pelas junções sinápticas do modelo biológico, chamadas no neurônio artificial por pesos sinápticos, são representados por um conjunto de pesos $\mathbf{w} = \{w_1, w_2, \dots, w_n\}$. A soma ponderada dos sinais de entrada e seus respectivos pesos sinápticos, ou seja, $\sum_{i=1}^n x_i \times w_i$, resulta em um potencial de ativação, denotado por u , e representado no neurônio biológico pela saída do corpo celular que gera o sinal de saída propagado pelo axônio. Esse potencial passa por uma função de ativação (f) e resulta em uma saída representada por y no neurônio artificial. Ou seja, $y = f(u)$ (MCCULLOCH; PITTS, 1943).

Figura 5 – Arquitetura de um neurônio artificial



Fonte: SILVA; ZUBEN, 1998.

Em 1958 Rosenblat complementou o conceito introduzindo o Perceptron, um classificador linear (feedforward) que a partir do reconhecimento dos erros identificados no processo de treinamento inicia-se um processo de adaptação dos pesos sinápticos convergindo pouco a pouco para o aumento da precisão do algoritmo de classificação a cada exemplo fornecido pelo conjunto de treinamento da rede neural. (FRANK ROSENBLATT, 1958). Em 1962, Widrow e Hoff apresentaram o algoritmo conhecido como *Adaline* (do inglês *Adaptive Linear Element*), este algoritmo introduziu uma mudança na atividade da camada de saída da rede que deixou de atuar apenas com classificações binárias e começou a classificar múltiplos rótulos pré-estabelecidos por meio de funções lineares (WIDROW; HOFF, 1962). Posteriormente em 1969, Minsky e Papert realizaram uma análise rigorosa dos potenciais e limitações desta arquitetura, essa análise contribuiu para os próximos passos na evolução deste paradigma computacional. Em 1982 Hopfield publicou o conceito de redes neurais recorrentes, seguido por Kohonen que no mesmo ano introduziu o conceito de redes neurais auto gerenciáveis (MINSKY; PAPERT, 1969) (HOPFIELD, 1982) (KOHONEN, 1982).

Os modelos apresentados demonstraram o grande potencial das redes neurais. Entretanto sua aplicação em problemas mais complexos, como por exemplo, classificação de imagens, arquivos de áudio e vídeos demandava um esforço muito grande de pré-processamento na preparação do conjunto de dados para o treinamento da rede. Em busca de potencializar a aplicação das redes neurais em problemas como os exemplos citados, surgiu

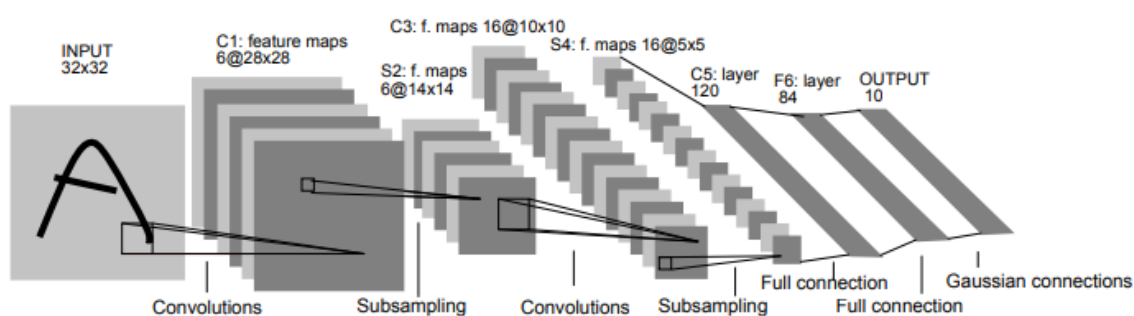
um novo tipo de rede conhecido como redes neurais profundas. De acordo com LeCun e colaboradores (1998), as redes profundas se assemelham bastante as redes Perceptron de múltiplas camadas, porém, com um número maior de camadas ocultas. As camadas ocultas são capazes de diferenciar imagens extraindo características que possibilitam a realização de tarefas de classificação com pouco ou nenhum esforço de pré-processamento (LECUN et al., 1998). Esta nova arquitetura conhecida como LeNet foi apresentada em seu trabalho, onde de forma pioneira, o conceito de redes neurais profundas foi introduzido para construir um algoritmo de reconhecimento de dígitos, conforme ilustrado na Figura 6.

As redes profundas em geral possuem dezenas de camadas e com o seu surgimento as arquiteturas desenvolvidas anteriormente começaram a ser denominadas redes neurais rasas, pois geralmente mesmo as redes de “múltiplas camadas” não possuem mais de duas ou três camadas ocultas.

Atualmente as redes profundas vêm se tornando cada vez mais populares em razão da sua capacidade de solução de problemas complexos como a classificação de imagens médicas por exemplo.

A Figura 6 demonstra a arquitetura de uma rede neural convolucional, este nome é originário da operação matemática denominada convolução. A imagem permite observar a distribuição de responsabilidades entre as diferentes camadas da rede.

Figura 6 - Arquitetura Lenet-5 aplicada ao reconhecimento de dígitos



Fonte: LECUN et al., 1998.

Aprendizagem profunda se difere das demais aprendizagens, no que tange ao nível de abstração que é capaz de obter enquanto extrai as características dos objetos que analisa por entre suas múltiplas camadas. Entretanto, esta abstração torna uma tarefa árdua definir a arquitetura ideal e ajustar os pesos de modo apropriado para aplicação em cada problema específico.

4.2 Paradigmas de Aprendizagem em RNAs

O processo de aprendizagem é a principal característica dos modelos computacionais baseados em redes neurais artificiais. O algoritmo utilizado no processo de aprendizagem tem grande influência no desempenho da rede. A definição geral de aprendizagem foi apresentada por Haykin (2008) como o processo adaptativo imposto aos parâmetros livres de uma rede neural durante a exposição da rede a estímulos externos (HAYKIN, 2008).

O aprendizado da rede neural ocorre por meio de um processo treinamento, durante o treinamento a rede neural interage com os exemplos do conjunto de dados diversas vezes, este ciclo de interações sucessivas entre a rede e os elementos do conjunto de dados e a rede neural é denominado época. O número de épocas ideal para o treinamento pode variar de acordo com a arquitetura escolhida e com a complexidade do problema em questão. As regras de aprendizagem são responsáveis por definir o ajuste dos pesos entre as épocas do treinamento da rede neural (BASHEER; HAJMEER, 2000). Segundo Haykin (2008), as regras de aprendizagem podem ser definidas, basicamente, em dois paradigmas: Aprendizado supervisionado e aprendizado não supervisionado (HAYKIN, 2008).

4.2.1 Aprendizado Supervisionado

Este paradigma considera uma entidade detentora do conhecimento que se refere às respectivas saídas pré-estabelecidas de cada elemento de entrada do conjunto de treinamento da rede, esta entidade é conhecida como Professor. Os elementos de entrada são representados em geral por vetores compostos por atributos do elemento a ser classificado. Ao longo do processo de treinamento, o Professor e a rede neural interagem com os elementos do conjunto de dados. A cada interação, em virtude do conhecimento prévio das saídas de cada entrada, o professor é capaz de fornecer à rede neural uma resposta desejada para o respectivo elemento do conjunto de dados. A resposta do professor representa a ação esperada pela rede neural (HAYKIN, 2008).

Os parâmetros da rede são ajustados mediante a influência combinada do vetor que representa os atributos do elemento e do sinal de erro. O sinal de erro é definido como a diferença entre a resposta desejada e a resposta real da rede. Este ajuste é realizado com o objetivo de ensinar a rede neural a obter a mesma saída do professor no maior número possível de elementos do conjunto de dados (HAYKIN, 2008).

A média de todos possíveis exemplos de entrada-saída é denominada superfície de erro. A saída real da rede é representada como um ponto na superfície do erro, para que a rede

melhore seu desempenho ao longo das épocas de treinamento o ponto da superfície de erro deve ser reduzido sucessivamente em direção a um ponto mínimo. A redução do ponto de erro ocorre por meio de algoritmos de aprendizado. Os principais algoritmos de aprendizado serão apresentados no subcapítulo 4.3.

Esta técnica de aprendizado é a mais comumente utilizada no desenvolvimento de algoritmos de classificação de imagens médicas como observado nos trabalhos correlatos elencados neste estudo.

4.2.2 Aprendizado Não Supervisionado

No paradigma de aprendizado não supervisionado não existe o conceito do professor utilizado na abordagem supervisionada. Em outras palavras, não há exemplos previamente rotulados da aplicação a ser aprendida pela rede. Este paradigma subdivide-se em dois grupos (HAYKIN, 2008):

a) Aprendizagem Auto Organizada

A abordagem denominada aprendizagem Auto Organizada, se caracteriza pela ausência completa de um supervisor para o processo de aprendizado. Ou seja, durante o treinamento os sinais de entrada são transmitidos para a rede sem a resposta esperada na saída do processamento. O aprendizado é avaliado de forma independente, com foco na qualidade da representação do problema que a rede deve aprender. Conseqüentemente, os parâmetros livres da rede são adaptados com o objetivo de alcançar a harmonia entre as regularidades estatísticas dos elementos de entrada e a saída da rede. Em termos práticos, este paradigma realiza uma concorrência de aproximação estatística entre os neurônios para identificar a conexão em maior harmonia com o elemento de entrada (HAYKIN, 2008).

Quando aplicada em problemas que possuem informações com altos índices de regularidade a rede é capaz de obter bons resultados em sua aprendizagem como demonstram os estudos realizados por Andrade e colaboradores em 1993 quando a técnica foi aplicada para agrupar estruturas de proteína em espectros de dicroísmo circular (ANDRADE et al, 1993).

Os autores Libbrecht e Noble (2015) em uma revisão da literatura sobre a aplicação de inteligência artificial na área de medicina de precisão destacam a importância da aplicação desta técnica quando se procura aprender mais sobre um determinado conjunto de dados ainda inexplorados (LIBBRECHT; NOBLE, 2015). Além da possibilidade de mapear similaridades

e padrões entre diferentes grupos conhecidos abstraindo das classificações pré-estabelecidas como nos estudos conduzidos por Hemmaplardh e colaboradores (2007) e de Hoffman e colaboradores (2012) onde a técnica foi aplicada no mapeamento de genomas humanos (HEMMAPLARDH et al, 2007) (HOFFMAN et al, 2012). Outro exemplo da aplicação desta técnica é o estudo conduzido por Hilas e Mastorocostas em 2008 quando construíram um algoritmo baseado em aprendizado não supervisionado somado a técnicas de distância euclidiana e correlações para identificar fraudes na obtenção de serviços de Telecomunicação (HILAS; MASTOROCOTAS, 2008).

b) Aprendizagem por reforço

Segundo Haykin (2008) a aprendizagem por reforço, se baseia em entradas e saídas provenientes de uma interação contínua com as saídas anteriores da rede. O propósito desta interação continua é que a rede aprenda com estímulos de reforço em longo prazo, ou seja, a rede observa uma sequência temporal de estímulos recebidos minimizando a função de ajuste imediato dos pesos e, considerando uma maior importância ao resultado acumulado das ações em uma sequência de etapas ao invés de simplesmente o resultado imediato. Essa abordagem é fundamental em situações onde certas decisões tomadas pela rede podem apresentar um resultado imediato positivo, mas não se provarem as melhores opções quando avaliadas em uma perspectiva de longo prazo. A função do sistema de aprendizagem é descobrir essas ações e retroalimentar a rede para embasar decisões futuras. O aprendizado por reforço é difícil de ser executado por duas razões principais (HAYKIN, 2008):

- Não existe o conceito do professor para fornecer uma resposta desejada a cada etapa do processo de aprendizagem.
- Atraso incorrido na geração do sinal de reforço primário, esta abordagem implica que o algoritmo de aprendizagem deve resolver um problema de atribuição de créditos temporais. Ou seja, o modelo deve ser capaz de atribuir crédito e culpar individualmente cada ação na sequência de etapas de tempo que levaram ao resultado final, enquanto o reforço primário só pode avaliar o resultado.

Apesar dessas dificuldades, o aprendizado por reforço fornece a base para o sistema de aprendizagem interagir com o ambiente desenvolvendo assim a capacidade de aprender a executar uma tarefa prescrita apenas com base nos resultados de sua experiência resultante da

interação (HAYKIN, 2008). Um exemplo da aplicação de aprendizado por reforço é o estudo conduzido por Gottipati e colaboradores (2018). Neste estudo os pesquisadores aplicaram a técnica para criar um mecanismo de inteligência artificial especializado em jogar xadrez. O projeto utilizou processamento distribuído e reforço de longo prazo para treinar o modelo considerando o efeito de cada movimentação das peças no desfecho do jogo (GOTTIPATI et al, 2018).

4.3 Algoritmo de aprendizagem por correção de erro

Segundo Haykin (2008), o conceito de aprendizagem pode ser definido como a modificação adaptativa dos parâmetros de uma rede neural por meio dos estímulos originários do ambiente. O modelo de aprendizagem, por sua vez, é o processo intrínseco por onde os ajustes dos parâmetros acontecem (HAYKIN, 2008).

A abordagem de aprendizado baseada em ajustes de erro é a técnica a qual o sinal de erro proveniente de um reforço negativo pela divergência entre a saída esperada e a saída real da rede é utilizado para ajustar os pesos sinápticos em ordem de melhorar o desempenho da rede para as próximas análises. Esse ajuste nos pesos ocorre por meio de um processo de retro-propagação que pode ser dividido em duas fases (SILVA; ZUBEN, 1998):

Feedforward: Feedforward pode ser definido como a fase de propagação do sinal funcional entre as camadas da rede. Nesta etapa, os valores provenientes do ambiente são obtidos como o sinal de entrada e modificados pela rede enquanto transitam pelas camadas até gerar o sinal de saída. Caso a saída da rede seja a esperada pelo professor, os valores dos pesos sinápticos serão mantidos, caso contrário, será ativado o processo de retro-propagação do erro (Backpropagation) (SILVA e ZUBEN, 1998).

Backpropagation: O algoritmo Backpropagation inicialmente apresentado pelos autores Rumelhart, Hinton e Williams (1988), tem o intuito de proporcionar as redes neurais de múltiplas camadas uma forma de extrair conhecimento por meio de seus erros de classificação. Logo, o algoritmo é definido como a fase de retro-propagação do erro (RUMELHART; HINTON; WILLIAMS, 1988). Esta etapa do processo de aprendizado ocorre quando a saída da rede é diferente da esperada pelo professor. Quando este erro acontece, o sinal de erro é calculado de modo a ajustar a distancia entre a resposta da rede e a saída esperada. Após a conclusão do

calculado o ajuste é propagado da camada de saída para a camada de entrada modificando os pesos sinápticos por todas as camadas ocultas em ordem de ajustar a classificação da rede (JAIN; MAO, 1996).

Existem várias abordagens para construção de algoritmos de aprendizado. Entretanto, uma das abordagens mais bem-sucedidas é a aprendizagem baseada em gradientes (LECUN et al., 1998).

4.3.1 Gradiente Descendente

Proposto por Denis e Schanabel (1983) o Gradiente Descendente (GD) é utilizado para minimizar o risco empírico, definido como a média da taxa de erro nos elementos do conjunto de treinamento, por meio da atualização dos pesos da rede neural. O cálculo do gradiente é expresso pela equação 1 (DENIS; SCHANABEL, 1983):

$$w_{t+1} = w_t - \gamma \frac{1}{n} \sum_{i=1}^n \nabla_w Q(z_i, w_t) \quad (1)$$

Onde, w_{t+1} representa o valor do peso na próxima interação da rede, ajustado com base no peso atual denotado por w_t ; e γ representa um valor escalar aplicado ao cálculo do ajuste que deve ser adequadamente escolhido a partir de suposições de regularidade, quando a estimativa inicial de w estiver perto o suficiente do ideal, e quando o ajuste γ é suficientemente pequeno, este algoritmo consegue atingir uma convergência linear (DENIS e SCHANABEL, 1983). O número total de atributos da entrada aplicados na ponderação é representado por n . z representa um par obtido pela entrada da rede (x) e a saída esperada (y), ou seja, $z = (x, y)$. Portanto a função de perda é determinada por $Q(z_i, w_t)$.

Bottou (2010) sugere que o gradiente obteria um melhor desempenho substituindo o ajuste escalar γ por uma matriz positiva definida por Γ_t . O ajuste é expresso pela equação 2.

$$w_{t+1} = w_t - \Gamma_t \frac{1}{n} \sum_{i=1}^n \nabla_w Q(z_i, w_t) \quad (2)$$

Segundo o autor essa variação denominada descida de gradiente de segunda ordem é uma variante capaz de atingir convergência quadrática se o valor inicial de w for suficientemente próximo do gradiente ótimo de segunda ordem. Quando o custo é quadrático e a matriz de escalonamento Γ é exata, o algoritmo atinge o ótimo depois de uma única iteração (BOTTOU, 2010).

4.3.2 Gradiente Descendente Estocástico

O Gradiente Descendente Estocástico (GDE) é uma simplificação do Gradiente Descendente baseado no teorema de Robbins-Siegmund que fornece os meios para estabelecer uma convergência segura em condições moderadas (Robbins e Siegmund, 1971) (BOTTOU, 2010). Enquanto GD calcula o erro empírico pela ponderação do erro entre todos os atributos de entrada a cada iteração, o GDE calcula esse gradiente com base em um único exemplo escolhido aleatoriamente, este atributo é representado por (z_t) na equação 3 (BOTTOU, 2010).

$$w_{t+1} = w_t - Y_t \nabla_w Q(z_t, w_t) \quad (3)$$

O processo estocástico depende dos atributos aleatoriamente escolhidos em cada iteração. Este procedimento simplificado pode introduzir um ruído ao aprendizado em razão da escolha randômica do atributo. Entretanto, apesar do ruído, a descida do gradiente estocástico aperfeiçoa diretamente o risco empírico esperado (BOTTOU, 2010).

A convergência do gradiente estocástico foi estudada extensivamente na literatura de aproximação estocástica e resultados de convergência geralmente exigem ganhos decrescentes que satisfaçam as condições $\sum_t Y_t^2 < \infty$ e $\sum_t Y_t = \infty$ (BOTTOU, 2010).

A segunda ordem do gradiente estocástico foi proposta por Bordes e colaboradores (2009), o calculo é expresso pela equação 4.

$$w_{t+1} = w_t - Y_t \Gamma_t \nabla_w Q(z_t, w_t) \quad (4)$$

Segundo os autores, essa modificação não reduz o ruído estocástico e, portanto, não melhora significativamente a variância do peso. Entretanto, o erro empírico é reduzido.

4.4 Transferência de Aprendizado

O conceito de transferência de aprendizado é definido como o aproveitamento do aprendizado obtido por uma rede neural a partir do treinamento realizado por meio de um conjunto de dados, denominado conjunto de base, no reconhecimento de características genéricas, presentes em qualquer conjunto de dados composto por imagens para aprimorar os resultados obtidos no treinamento de um segundo conjunto de dados denominado conjunto de destino. Devido à transferência de aprendizado ser relativa somente à extração de características genéricas, o conjunto de destino não precisa ser necessariamente composto por imagens semelhantes às imagens presentes no conjunto de base (YOSINSKI et al., 2014).

Quando o conjunto de dados de destino é significativamente menor que o conjunto de base, a transferência de aprendizado pode ser uma ferramenta para aprimorar o resultado e reduzir o *overfitting* (SERMANET et al., 2014).

As arquiteturas de redes neurais profundas, quando treinadas para classificação de imagens, geralmente atribuem responsabilidades semelhantes à primeira camada. Independente da natureza das imagens que compõe o conjunto de dados ou da densidade da rede. É comum que a primeira camada aprenda características relacionadas à textura e coloração das imagens. Esta característica das redes profundas é tão frequente que qualquer comportamento diferente pode indicar que os parâmetros da rede não foram configurados de maneira apropriada (YOSINSKI et al., 2014).

A última camada das redes neurais se comporta de maneira oposta, variando consideravelmente de acordo com o conjunto de dados e cada unidade de saída se torna específica para uma determinada classe alvo. A partir destas observações é possível intuir que o comportamento das camadas deixa de ser generalizado e se torna específico para o problema que está sendo abordado à medida que avançamos das primeiras para as últimas camadas da rede (YOSINSKI et al., 2014).

Estes conceitos possibilitam adotar diferentes estratégias para inicializar os pesos da rede neural no processo de treinamento. Algumas das principais abordagens são descritas como segue:

4.4.1 Inicialização Randômica (*IR*)

A inicialização randômica consiste em iniciar o treinamento da rede com valores de pesos aleatórios e permitir que os pesos de todas as camadas da rede sejam adaptados ao longo das épocas do treinamento. Como os pesos são inicializados de forma aleatória, esta abordagem não aplica transferência de aprendizado e, portanto, não requer a utilização de um conjunto de base (PITTARAS et al., 2017).

4.4.2 Extensão de Modelo com Reinicialização de Pesos (*EMRP*)

Esta abordagem de inicialização dos pesos aplica a transferência de aprendizado. Após o treinamento do conjunto de base, os pesos treinados das camadas responsáveis pela extração de características são copiados e o treinamento do conjunto de destino é iniciado utilizando os valores de pesos provenientes do treinamento do conjunto de base. Os pesos das camadas restantes da rede de destino são inicializados com valores randômicos e treinados para a tarefa em questão. Nesta abordagem os erros são retropropagados a todas as camadas da rede. Portanto, todas as camadas se adaptam ao conjunto de destino (PITTARAS et al., 2017).

4.4.3 Extensão de Modelo com Extratores Fixos (*EMEF*)

Esta abordagem de inicialização dos pesos assim como a *EMRP*, também aplica a transferência de aprendizado. Entretanto, nesta abordagem os erros são retropropagados somente entre as camadas responsáveis pela classificação, mantendo os pesos das camadas utilizadas na extração de características estáticos ao longo do treinamento do conjunto de destino (PITTARAS et al., 2017).

4.5 Funções de Ativação

Algumas das primeiras propostas de funções de ativação, usadas para a computação em redes neurais foram exploradas por Elliott (1993), quando ele estudou o impacto das Funções de ativação (FAs) em redes neurais (ELLIOTT, 1993).

Funções de ativação podem ser lineares ou não lineares. Entretanto, em geral as funções trazem componentes de não linearidade em sua saída. Logo, a presença de características dinâmicas nas propriedades envolvidas na definição do estado de ativação tem

como consequência a aplicação de equações diferenciais. Caso contrario, não poderiam funcionar durante a retro propagação dos sinais das redes neurais (SILVA; ZUBEN, 1998).

As funções de ativação são utilizadas para calcular a soma ponderada das entradas e vieses, mais conhecidos como *bias*. *Bias* são parâmetros aplicados na decisão de ativação dos neurônios. Arquiteturas de múltiplas camadas, em especial as redes profundas, são compostas por várias camadas de processamento. Cada camada envolve operações lineares e não lineares aplicadas em conjunto para resolver um determinado problema.

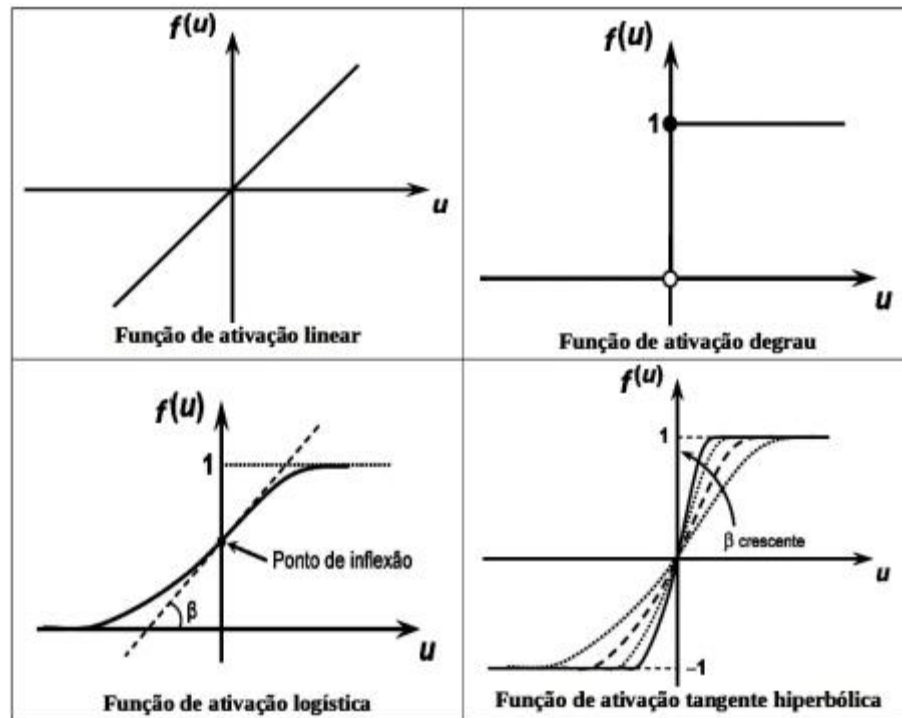
As FAs são responsáveis por realizar o processamento dos parâmetros livres calculando a entrada da camada subsequente e saída da rede neural. (NWANKPA et al., 2018).

Os ajustes calculados pelos gradientes são geralmente menores que 1. Conseqüentemente com a multiplicação sucessiva destes resultados, o valor torna-se cada vez menor e tende à zero, reduzindo a ação do gradiente e comprometendo a acurácia do algoritmo de classificação. Por outro lado, se os valores forem maiores que 1, a multiplicação sucessiva aumentará os valores e o gradiente tende ao infinito, ultrapassando os limites do gradiente. Assim, as FAs mantêm os valores desses gradientes em intervalos moderados (NWANKPA et al., 2018).

Em 2018 um estudo conduzido por NWANKPA e colaboradores apresenta evidências de diversos trabalhos que demonstram a importância de se utilizar a função de ativação apropriada para elevar a acurácia obtida no treinamento das redes neurais artificiais (NWANKPA et al., 2018).

Algumas das funções de ativação mais utilizadas nas arquiteturas de redes neurais tradicionais são apresentadas na Figura 7, onde β é uma constante real associada ao nível de inclinação da função logística relacionada ao ponto de inflexão.

Figura 7 - Principais funções de ativação das redes neurais rasas



Fonte: SILVA; SPATTI; FLAUZINO. 2010.

Apesar do aceitável desempenho das funções de ativação apresentadas nas arquiteturas de redes neurais tradicionais, as redes neurais profundas em geral apresentam melhores resultados quando utilizadas com as funções de ativação descritas a seguir:

4.5.1 Função de ativação Logística

A função de ativação Logística é uma função não-linear usada principalmente em redes neurais feedforward. É uma função real e, portanto, definida para valores de entrada reais e derivados positivos. A função Sigmoidal se dá por meio da equação (NWANKPA et al., 2018):

$$f(u) = \left(\frac{1}{1 + \exp^{-u}} \right)$$

(5)

A função de ativação Logística também está presente nas camadas de saída de arquiteturas de redes neurais profundas, e em redes tradicionais, também conhecidas como redes rasas. Além disso, Glorot e Bengio, 2010 sugerem que a função de ativação Logística deve ser evitada ao inicializar a rede neural a partir de pesos sinápticos com valores aleatórios. No entanto, a função sofre algumas desvantagens, tais como gradientes reduzidos excessivamente durante a retropropagação de níveis mais profundos, saturação de gradiente, convergência lenta e saída centrada diferente de zero. Outras formas de função de ativação, incluindo a função Softmax, foram propostas para remediar algumas dessas desvantagens sofridas pela função de ativação Logística (NWANKPA et al., 2018).

4.5.2 Função de ativação Softmax

A função Softmax é outro tipo de função de ativação usada na computação neural. Essa função é utilizada para calcular a probabilidade de distribuição de um vetor de números reais. A função Softmax produz uma saída que é um intervalo de valores entre 0 e 1, com a soma das probabilidades igual a 1. A função Softmax é calculada usando a equação 6 (NWANKPA et al., 2018):

$$f(u_i) = \frac{\exp(u_i)}{\sum_j \exp(u_j)} \quad (6)$$

A função Softmax é usada geralmente em modelos multi-classe onde retorna probabilidades de cada classe, com a classe alvo tendo a maior probabilidade. A função Softmax aparece principalmente em as camadas de saída das arquiteturas de aprendizagem profunda. A principal diferença entre função Logística e a Softmax é que a função Logística é usada na classificação binária enquanto o Softmax é usado para tarefas de classificação multivariada (NWANKPA et al., 2018).

4.5.3 Rectified Linear Unit (ReLU)

A função ReLU foi proposta originalmente por Nair e Hinton em 2010, e desde então, tem sido a função de ativação mais utilizada para aplicações de aprendizagem profunda (NAIR; HINTON, 2010).

A ReLU é conhecida por ser uma função de ativação de baixo custo computacional, que apesar da simplicidade das operações realizadas obtém bons resultados e, portanto é

amplamente utilizada (NWANKPA et al., 2018). A função ReLU possui grande capacidade de generalização da aprendizagem profunda. Estudos demonstram resultados superiores quando comparada a outras FAs como regressão logística e tangente hiperbólica (DAHL; SAINATH; HINTON, 2013) (ZEILER et al., 2013).

O ReLU representa uma função quase linear e, portanto, preserva as propriedades dos modelos lineares que os tornam fáceis de otimizar com métodos do Gradiente Descendente, executando uma operação de limite para cada elemento de entrada em que valores menores que zero são definidos como 0, a função ReLU é expressa pela equação 7:

$$f(u) = \max(0, u) = \begin{cases} u_i, & \text{se } u_i \geq 0 \\ 0, & \text{se } u_i < 0 \end{cases} \quad (7)$$

Esta função retifica os valores das entradas menores que zero, forçando-as a zero e eliminando o problema observado nas funções de ativação *Softmax* e regressão logística. A função ReLU é comumente utilizada nas camadas ocultas das redes neurais profundas com outra função de ativação aplicada nas camadas de saída da rede. Esta abordagem apresentou resultados expressivos no estudo realizado por Krizhevsky, Sutskever e Hinton (2012) no desafio LSVRC-2012 no qual foi apresentada a arquitetura AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

Uma das limitações conhecidas desta função de ativação que é facilmente superada em comparação com a função de ativação Logística é o sobreajuste, também conhecido como *overfitting*. Este efeito é ocasionado pela adaptação excessiva do modelo ao conjunto de dados de treinamento reduzindo a capacidade de generalização do algoritmo (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Embora a técnica denominada *dropout*, que consiste em desconectar conexões dos neurônios randomicamente, tenha sido proposta para solucionar esta fragilidade, ela apenas ameniza este efeito (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Outra limitação significativa desta técnica que pode causar fragilidades no treinamento é a eventual redução das atualizações dos pesos que ocasiona também o desligamento de neurônios e conseqüentemente a não ativação dos pesos no futuro impactando a aprendizagem. Para resolver este problema a arquitetura *Leaky ReLU* foi proposta (NWANKPA et al., 2018).

4.5.4 Leaky Rectified Linear Unit (LReLU)

A função de ativação denominada LReLU foi proposta em 2013 por Maas, Hannun e Ng como uma adaptação que introduz uma constante negativa α para a ReLU manter as atualizações de pesos ativas durante todo o processo de propagação. O parâmetro α foi introduzido de modo que os gradientes não sejam zero a qualquer momento durante o treinamento (MAAS; HANNUN; NG, 2013). A função LReLU calcula o gradiente com um valor constante muito pequeno para o gradiente negativo α no intervalo de 0,01, assim, a LReLU é calculado por meio da equação 8:

$$f(u) = \alpha u + u = \begin{cases} u & \text{se } u > 0 \\ \alpha u & \text{se } u \leq 0 \end{cases} \quad (8)$$

O LReLU tem um resultado similar aos resultados obtidos pela função ReLU, com uma exceção de que ele possui gradientes diferentes de zero ao longo de toda a duração. Portanto essa variação não resultou em melhorias significativas dos resultados, exceto na dispersão quando comparado com as funções padrão ReLU e tanh (MAAS; HANNUN; NG, 2013). A validação da função LReLU foi validada em um conjunto de dados de reconhecimento automático de voz.

4.5.5 Parametric Rectified Linear Units (PReLU)

A função ReLU paramétrico mais conhecida como PReLU é outra variante da ReLU proposta por He e colaboradores (2016). O PReLU apresenta uma parte negativa da função, apresentando o aprendizado adaptativo enquanto a parte positiva é linear. O PReLU é expressado pela equação 9 (HE et al., 2016):

$$f(u_i) = \begin{pmatrix} u_i, & \text{se } u_i > 0 \\ a_i u_i, & \text{se } u_i \leq 0 \end{pmatrix} \quad (9)$$

a_i é o parâmetro de controle de declive negativo e seu aprendizado durante o treinamento com retropropagação. Se o termo $a_i = 0$. A função tem o comportamento equiparado a ReLU (HE et al., 2016).

O desempenho do PReLU foi melhor que o ReLU no reconhecimento de imagens em larga escala. Os resultados do PReLU foram os primeiros a superar o desempenho humano no

desafio ILSVRC2014 de reconhecimento visual. O algoritmo desenvolvido ficou posteriormente conhecido como GoogLeNet (HE et al., 2016).

4.5.6 Exponential Linear Units (ELUs)

Proposta por Clevert, Unterhiner e Hochreiter (2015), as unidades lineares exponenciais (ELUs) é uma função de ativação utilizada para acelerar o treinamento das redes neurais profundas. A principal vantagem da ELU é reduzir o problema do gradiente de fuga usando identidade para valores positivos e também melhorar as características de aprendizagem. A função possui valores negativos que permitem aproximar a ativação da unidade para mais próximo de zero, reduzindo assim a complexidade computacional e melhorando a velocidade do aprendizado (CLEVERT; UNTERHINER; HOCHREITER, 2015).

O ELU representa uma boa alternativa à ReLU, pois diminui as alterações de polarização, empurrando a ativação média para zero durante o treinamento.

A unidade linear exponencial (ELU) é expressa por:

$$f(u) = \begin{cases} u, & \text{se } u > 0 \\ \alpha \exp(u) - 1, & \text{se } u \leq 0 \end{cases} \quad (10)$$

Nesta equação α representa os parâmetros da função ELU que controla o ponto de saturação para entradas negativas que geralmente é definido como. Os ELUs têm um patamar de saturação claro em seu regime negativo, aprendendo representações mais robustas, e oferecem uma aprendizagem mais rápida e maior capacidade de generalização em comparação com a ReLU e LReLU com estrutura de rede específica, especialmente acima de cinco camadas e garante resultados de última geração em comparação com as variantes ReLU. No entanto, uma limitação crítica do ELU é a não centralização dos valores em zero. A variação conhecida como SELU foi proposta para resolver esse problema (CLEVERT; UNTERHINER; HOCHREITER, 2015).

4.5.7 Scaled Exponential Linear Unit (SELU)

As unidades lineares exponenciais escalonadas (SELU) são uma variante da ELU, proposta por Klambauer e colaboradores em 2017. A SELU foi introduzida como uma rede neural com a capacidade de se normalizar automaticamente. Essa função de ativação possui uma variância que induz normalização. A variância unitária quando propagada através de

múltiplas camadas durante o treinamento da rede, a torna mais adequada para aplicações em aprendizagem profunda e em virtude da forte regularização, aprende recursos robustos de forma eficiente (KLAMBAUER et al., 2017).

A função SELU é expressa pela equação:

$$f(u) = \tau \begin{pmatrix} u, & \text{se } u > 0 \\ \alpha \exp(u) - \alpha, & \text{se } u \leq 0 \end{pmatrix} \quad (11)$$

Sendo τ o fator de escalonamento, o valor aproximado do parâmetro da função SELU é $\alpha \approx 1.6733$.

A função SELU não é afetada por problemas graduais de redução ou aumento excessivo dos gradientes. Os autores relataram que eles permitem a construção de mapeamentos com propriedades que levam as redes neurais auto-normalizantes que não podem ser derivadas usando ReLU, Logística, LReLU e até Tangente Hiperbólica Escalada.

4.6 Arquitetura de RNAs

A forma pela qual os neurônios de uma rede neural artificial se conectam possui grande relação com o algoritmo de aprendizagem utilizado durante a etapa de treinamento, e por sua vez as conexões criadas entre os neurônios e a arquitetura utilizada estão intimamente ligados ao desempenho do algoritmo de classificação (SILVA; ZUBEN, 1998).

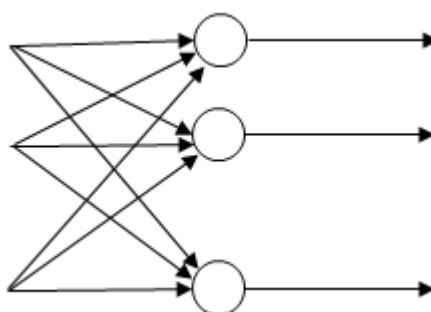
As redes neurais rasas são classificadas por Jain e Mao (1996) em apenas dois tipos de arquiteturas de RNAs: redes feedforward e redes recorrentes (JAIN; MAO, 1996). Entretanto, Haykin (2008) divide a arquitetura de uma RNA rasa em três tipos: redes feedforward de camada simples, redes feedforward de camada múltiplas e redes recorrentes. Neste estudo consideramos estes três tipos de arquiteturas descritos a seguir (HAYKIN, 2008).

4.6.1 Redes feedforward de camada simples

As redes feedforward de camada simples são estruturas compostas por apenas duas camadas, uma primeira camada de entrada e uma segunda camada de neurônios que determina a saída da rede. Nesta arquitetura os sinais são propagados exclusivamente de forma progressiva entre as camadas. Ou seja, de modo acíclico. A definição camada única faz

referência à camada de saída que possui em si uma estrutura de neurônios. A entrada da rede apesar de ser uma camada, não possui neurônios. Conseqüentemente, nela não ocorrem os processos computacionais atrelados ao processamento da rede, apenas a captação das características fornecidas pelo vetor de entrada da rede (HAYKIN, 2008). Alguns exemplos de redes que aplicam esta arquitetura são as redes Perceptron e Adaline, o modelo pode ser observado na Figura 8 onde as entradas são vetores de características extraídas de um objeto a ser classificado e a saída, em geral binária, é à classe na qual o objeto foi rotulado.

Figura 8 - Rede feedforward de camada única



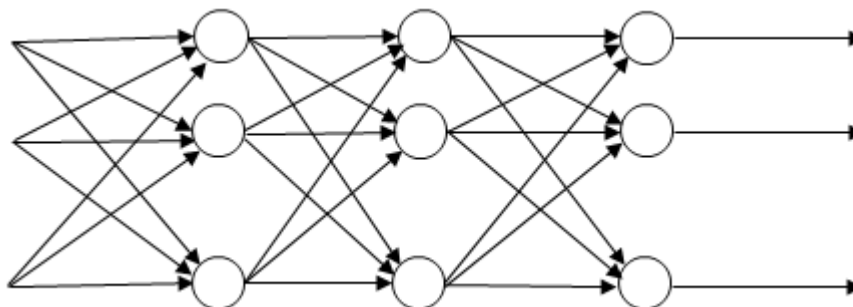
Fonte: Haykin, 2008.

4.6.2 Redes feedforward de camada múltipla

As redes feedforward de camada múltipla além das camadas de entrada e saída apresentam em sua arquitetura uma ou mais camadas intermediárias denominadas camadas ocultas. As camadas ocultas são compostas por estruturas neurais assim como a camada de saída. Este aumento no número de camadas proporciona uma elevação no potencial de processamento estatístico da rede sobre as características obtidas pela entrada da rede. Alguns exemplos de redes neurais que utilizam essa arquitetura são: Perceptron múltiplas camadas (*MLP* do inglês *Multilayer Perceptron*) e as redes de funções de base Radial (*RBF* do inglês *Radial Basis Function*). As redes *MLP* demonstram grande potencial de aplicação em diversos problemas, tais como, aproximação de funções, classificação, categorização e predição. Essa flexibilidade faz com que este tipo de rede seja aplicado em diversas áreas. Geralmente seu treinamento é realizado por meio da abordagem de aprendizado supervisionado utilizando algoritmos de retropropagação de erro baseados na regra de aprendizagem por correção (HAYKIN, 2008). A entrada da rede é um vetor de características

\mathbf{x} multiplicado por pesos \mathbf{w} e adicionado a um *bias* b . Cominando na equação *Perceptron*: $y = \mathbf{w} \times \mathbf{x}^T + b$. A Figura 9 demonstra um exemplo desta arquitetura.

Figura 9 - Rede feedforward de múltiplas camadas

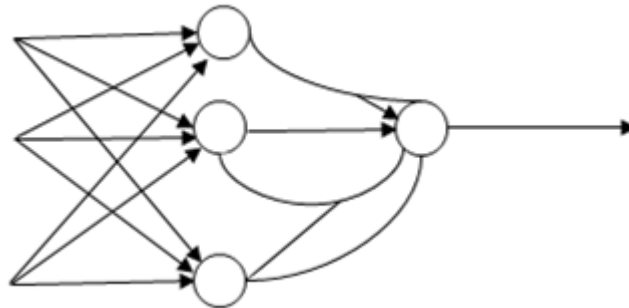


Fonte: Haykin, 2008.

4.6.3 Redes recorrentes

As redes recorrentes assim como as redes de camada múltipla possuem camadas intermediárias entre a entrada e a saída. Entretanto, uma nova característica a diferencia, esta arquitetura adiciona conexões ponderadas dentro das camadas, característica que não se faz presente em redes de múltiplas camadas tradicionais onde há correntes de alimentação somente entre camadas subsequentes. Esta característica tem por consequência a realimentação das saídas dos neurônios como sinais de entrada para outros neurônios e possibilita a rede realizar loops durante o processamento proporcionando a capacidade de armazenar informações e utilizar essas informações no processamento de novas entradas. Essa memória torna a rede mais assertiva para executar tarefas de processamento onde as entradas anteriores devem ser consideradas, como por exemplo, dados de série temporal. Por esta razão, as redes neurais profundas atuais são baseadas em redes recorrentes (HAYKIN, 2008). Um exemplo deste tipo de arquitetura são as redes Hopfield (HOPFIELD; TANK, 1986). A ilustração desta rede pode ser observada na figura 10 onde as entradas são transmitidas as camadas escondidas, propagadas a camada de saída e por fim, o resultado do processamento retroalimenta as camadas escondidas servindo de parâmetro de entrada para o processamento das entradas subsequentes.

Figura 10 - Rede recorrente



Fonte: Haykin, 2008

4.6.4 Redes Profundas

Diferente das redes neurais rasas as redes neurais profundas possuem diversas camadas com propósitos distintos. Os principais tipos de camadas presentes nas arquiteturas de redes neurais profundas são:

a) Camada Convolutional

Apresentada por Lecun e colaboradores em 1989, a camada convolutional recebe este nome por se basear na aplicação da função matemática denominada convolução (LECUN et al., 1998). A principal responsabilidade desta camada presente na maioria das arquiteturas de redes neurais profundas é extrair características dos vetores de entrada da rede para embasar a classificação. A função é expressa pela equação 12:

$$Y(i, j) = (K * X)(i, j) = \sum_m \sum_n X(i - m, j - n)K(m, n) \quad (12)$$

Nesta equação Y representa a saída e i e j as respectivas posições da matriz de entrada que serão submetidas à operação de convolução. Em geral a camada convolutional recebe como entrada uma matriz de dados multidimensional, por exemplo, uma imagem representada por X , o kernel K também representa uma matriz multidimensional de pesos composta pela ponderação da relevância das características presentes em X para a conversão do algoritmo de

aprendizado na classificação esperada. Em outras palavras, K é responsável por determinar a importância das características de X para a classificação da imagem. As matrizes apresentadas são conhecidas na literatura como tensores (GOODFELLOW; BENGIO; COURVILLE, 2016).

A camada convolucional apresenta três características importantes que influenciam drasticamente o custo computacional, a velocidade de aprendizado e a acurácia das redes neurais: Interações esparsas, Compartilhamento de parâmetros e Equivalência (GOODFELLOW; BENGIO; COURVILLE, 2016).

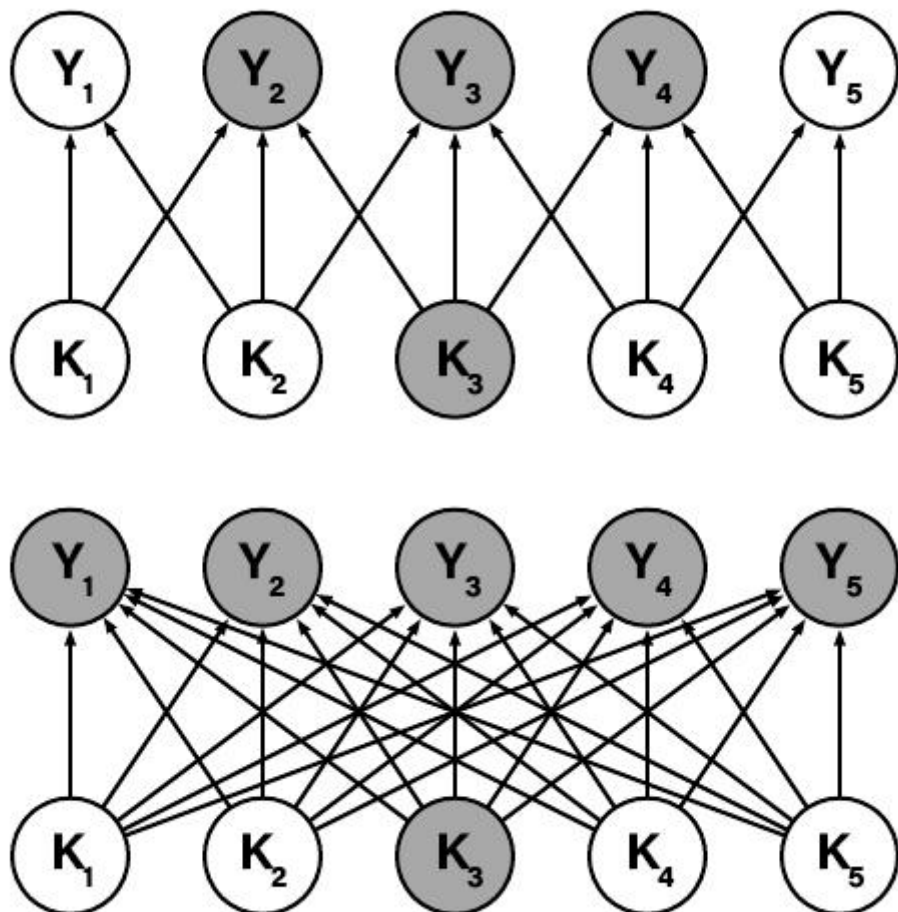
As camadas das redes neurais rasas em geral aplicam a multiplicação de matrizes por uma matriz de parâmetros com um parâmetro separado realizando a interação entre cada unidade de entrada e cada unidade de saída. Em outras palavras, cada unidade de saída interage com cada unidade de entrada. As camadas convolucionais por outro lado, executam interações esparsas entre os pesos da rede. Essa característica torna K consideravelmente menor que a entrada. Por exemplo, ao processar uma imagem, a entrada pode ter milhões de pixels, mas podemos detectar pequenos objetos de grande relevância para a classificação, como formas ou bordas específicas que ocupem poucas dezenas de pixels. Em termos matemáticos, as melhorias na eficiência costumam ser bastante representativas em razão da redução do número de operações. Considerando m o número total de parâmetros de entrada e n como o número total de parâmetros de saída, a multiplicação das matrizes ocorreria por $(m \times n)$ e os algoritmos na prática realizariam a multiplicação de todos os parâmetros $(m \times n)$. Se o número de conexões que cada saída pode ter para k for limitado, então na abordagem conectada de forma esparsa apenas os parâmetros k e n seriam multiplicados em tempo de execução $(k \times n)$.

Deste modo é possível reduzir o número de parâmetros e por consequência o volume de operações matemáticas realizadas pela rede. Os requisitos de memória necessários para o processamento também apresentam uma considerável redução. Além dos benefícios relacionados ao custo computacional, este aspecto tem o potencial de melhorar a acurácia do modelo em razão do aumento da eficiência estatística que se dá ao detectar e processar determinadas regiões de interesse (GOODFELLOW; BENGIO; COURVILLE, 2016).

O compartilhamento de parâmetros refere-se ao uso do mesmo parâmetro para mais de uma função em um modelo. Em uma rede neural rasa tradicional, cada elemento da matriz de pesos é usado uma vez ao calcular a saída da camada. Ou seja, após a multiplicação por um elemento da entrada o peso nunca é recuperado. As camadas convolucionais por outro lado armazenam instâncias de pesos, e o valor do peso aplicado a uma entrada está vinculado ao

valor de um peso aplicado em outro momento. Ou seja, cada item de K é usado em todas as posições de entrada. O compartilhamento de parâmetros é usado pela operação de convolução. Conseqüentemente, em vez de aprender um conjunto separado de parâmetros para cada local, apenas um conjunto é aprendido. Isso não afeta a propagação, a equação se mantém $(k \times n)$. Entretanto, reduz-se ainda mais os requisitos de armazenamento dos parâmetros do modelo contribuindo ainda mais com os requisitos de memória e eficiência estatística. A representação gráfica das conexões esparsas pode ser observada na figura 11 (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 11 – Conexões esparsas



Fonte: (GOODFELLOW; BENGIO; COURVILLE, 2016)

O princípio da Equivalência significa que quando um parâmetro de entrada de uma função é alterado, a saída deve ser modificada de forma equivalente. A camada convolucional tem o potencial de produzir uma linha do tempo registrando a presença de recursos nos

parâmetros de entrada. Em processamento de imagens, a convolução cria um mapa multidimensional, neste mapa determinados recursos são percebidos na entrada das camadas e se o objeto for movimentado na entrada, sua representação será movida proporcionalmente na saída. Essa propriedade é aplicada quando se identifica que um conjunto de pixels vizinhos pode ser determinante quando aplicado a vários locais de entrada. Conseqüentemente, torna-se uma boa estratégia detectar bordas na primeira camada da rede. As mesmas bordas tendem a aparecer regularmente em todos os lugares da imagem tornando interessante compartilhar os parâmetros em toda a imagem.

b) Camada de *Pooling*

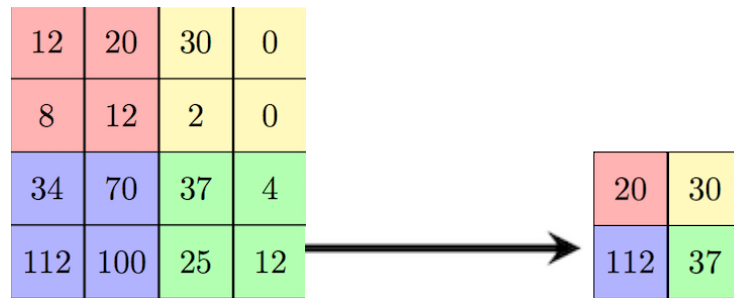
As camadas de *Pooling* estão presentes nas principais arquiteturas de redes neurais profundas. A principal finalidade das camadas de *pooling* é obter invariância espacial reduzindo a resolução dos mapas de características progressivamente à medida que os parâmetros são transmitidos entre as camadas da rede. Cada mapa de características reduzido corresponde a um mapa de características da camada anterior. Suas unidades combinam a entrada de uma matriz de $m \times m$ dimensões e gera matriz de saída de $n \times n$ dimensões. Conceitualmente as dimensões de $n \times n$ devem ser menores que $m \times m$. O tamanho da compactação aplicado pela camada é configurado durante a etapa de arquitetura da rede, existindo a possibilidade de sobreposição entre as matrizes de compactação (SCHERER; MÜLLER; BEHNKE, 2010).

Diversas abordagens são possíveis para atingir esta redução. Entretanto, as duas abordagens mais comuns são denominadas: Agrupamento máximo (do inglês *max pooling*), expressa na equação 13 e Subamostragem (do inglês *subsampling*) expressa na equação 14.

$$A_j = \max_{N \times N} A_i^{n \times n} h(n, n) \quad (13)$$

A função *Max pooling* (13) reduz a matriz de entrada denominada A_i para as proporções definidas na janela $h(n, n)$. O valor de cada célula da matriz de saída A_j é determinado pelo valor máximo da vizinhança selecionada na matriz de entrada. Conseqüente, o resultado é um mapa de características de menor resolução (SCHERER; MÜLLER; BEHNKE, 2010).

O efeito da função *max pooling* pode ser observado na figura 12, considerando o filtro de 2×2 pixels.

Figura 12 –Max Pooling

Fonte: SCHERER; MÜLLER; BEHNKE, 2010

$$A_j = \tan h \left(\beta \sum_{N \times N} A_i^{n \times n} + b \right)$$

(14)

A função *subsampling* (14) utiliza a média sobre as entradas e multiplica o valor por um escalar β e um *bias* b , gerando o resultado de forma não linear. Assim como na função de agrupamento máximo o resultado é um mapa de características de menor resolução (SCHERER; MÜLLER; BEHNKE, 2010).

c) Camada Totalmente Conectada

Como o nome indica em camadas totalmente conectadas todos os neurônios da camada anterior estão interligados a todos os neurônios da camada seguinte. A saída das camadas convolucionais e de agrupamento representam recursos de alto nível da imagem de entrada, esses recursos são transmitidos a as camadas totalmente conectadas que tem por objetivo utilizar esses recursos para classificar a imagem de entrada no conjunto de classes alvo com base no conjunto de dados de treinamento.

As camadas totalmente conectadas possuem o mesmo funcionamento das redes Perceptron de múltiplas camadas tradicionais. As camadas com esta configuração, em geral, compõe o ultimo bloco das redes neurais profundas, justamente pela função de realizar a classificação (SIMONYAN; ZISSERMAN, 2014).

Diversos estudos substituem este bloco de camadas das redes neurais profundas por outras ferramentas de classificação como as maquinas de vetores de suporte, por exemplo. Esta estratégia em alguns cenários pode melhor a acurácia da rede. Entretanto, no domínio de

classificação de lesões de pele, o estudo comparativo conduzido por Mahbod e colaboradores (2017) apontou um melhor desempenho para as camadas totalmente conectadas (MAHBOD et al., 2017).

Conforme o conceito introduzido anteriormente, as redes neurais profundas têm sua arquitetura definida entre outras características pela escolha dos blocos de camadas, pelo volume dos parâmetros transmitidos entre as camadas, pelo método de aprendizagem e pela função de ativação utilizada em cada camada. Algumas das principais arquiteturas de redes neurais profundas apresentadas na literatura são descritas a seguir.

4.6.4.1 LeNet

A arquitetura LeNet, conforme observado na Figura 6, é composta por sete camadas desconsiderando a camada de entrada. Nomeada por LeCun e colaboradores (1989) quando teve sua primeira implementação. Esta arquitetura foi desenvolvida inicialmente para identificar caracteres numéricos escritos à mão, logo é provável que a rede se deparasse com variações na posição de escrita e no tamanho dos caracteres, além de diferenças de coloração tanto no fundo quanto nos caracteres entre as amostras. Essa arquitetura obteve uma taxa de erro de 0,35% em um conjunto de dados de 60.000 caracteres (LECUN et al., 1998).

Todas as sete camadas possuem pesos que são ajustados ao longo do treinamento. A entrada é uma imagem de 32x32 pixels, dimensões suficientes para processar o maior caractere identificado no conjunto de dados de treinamento e validação da rede, que corresponde a 20x20 pixels centralizados em um espaço de 28x28 pixels.

A razão destas dimensões maiores do que o espaço ocupado pelos caracteres que compõe a região de interesse da imagem é potencializar a detecção das características distintivas, como por exemplo, caracteres posicionados nos cantos da imagem, serem reposicionados mais ao centro da matriz facilitam a ação das camadas convolucionais na detecção das características.

Na arquitetura LeNet os extratores de características da última camada convolucional C3 formam uma área central de 20x20 pixels na matriz de 32x32 que compõe a imagem de entrada. Os valores dos pixels de entrada foram normalizados de modo que o fundo onde estão escritos os caracteres apresentem uma cor branca uniforme correspondente ao valor de -0.1 e o caractere em primeiro plano obtenha a coloração preta no valor de 1.175. Esta normalização faz com que a entrada média seja de aproximadamente 0 e a variância aproximadamente 1, o que acelera o aprendizado (LECUN et al., 1998).

Após a camada de entrada, as camadas seguintes serão apresentadas como Cx quando convolucionais, Sx quando camadas de *pooling* baseadas em *subsampling* e Fx quando camadas totalmente conectadas. X representara índice da respectiva camada.

A Camada C1 possui seis mapas de recursos e cada unidade em cada mapa de características é conectada a um vizinho de 5x5 pixels na entrada. O tamanho dos mapas de recursos é 28x28 pixels o que impede a conexão da entrada de sair dos limites pré-estabelecidos. A camada C1 contém 156 parâmetros treináveis e 122.304 conexões.

A camada S2 também é composta por seis mapas de recursos. Entretanto, os mapas são representados por matrizes de 14x14 pixels. Cada unidade em cada mapa de características é conectada a um vizinho de 2x2 pixels do recurso correspondente em C1. As quatro entradas para uma unidade em S2 são adicionadas e multiplicadas por um coeficiente treinável e então adicionadas a um *bias* em treinamento. O resultado é processado por uma função Logística. Os campos de 2x2 não possuem sobreposições e, portanto, os mapas de recursos presentes em S2 são compostos pela metade do tamanho dos mapas de recursos em C1. A camada S2 é então composta por 12 parâmetros treináveis e 5880 conexões.

A Camada C3 é conectada a vários vizinhos compostos por 5x5 pixels em locais idênticos presentes nos subconjuntos dos mapas de recursos de S2.

Os primeiros seis mapas de recursos de C3 recebem as entradas de todos os subconjuntos de três mapas de recursos em S2 Os próximos seis mapas recebem a entrada de cada subconjunto contíguo de quatro. Os próximos três recebem a entrada de alguns subconjuntos descontínuos de quatro e o último mapa recebe entrada de todos os mapas de recurso de S2. A camada C3 é composta por 1.516 parâmetros e 151.600 e conexões.

A camada S4 é uma camada de subamostragem com 16 mapas de recursos de tamanho 5x5. Cada unidade em cada mapa de características é conectada a um vizinho de 2x2 no mapa de características correspondente em C3 semelhante ao que ocorre entre C1 e S2. S4 é composta por 32 parâmetros e conexões 2.000 conexões.

A Camada C5 possui 120 mapas de recursos. Cada unidade está conectada a um vizinho de tamanho 5x5 pixels em todos de mapas de recursos de S4, conseqüentemente o tamanho de C5 é 1x1. Essa abordagem de conexões se assemelha as camadas totalmente conectadas, entretanto C5 ainda é definida como uma camada convolucional porque se os parâmetros de entrada da rede fossem maiores e todo o restante das configurações se mantivesse constantes as dimensões dos mapas de recursos seriam maiores que 1x1. C5 é composta por 48.120 conexões.

Assim como nas camadas clássicas das redes neurais rasas, para F6 calcular um produto entre o vetor de entrada e o vetor de peso, um *bias* é adicionado. Essa soma ponderada denota u_i pela unidade i e então é propagada através de uma função de ativação logística (f) para produzir o estado da unidade i é expresso por y_i :

$$y_i = f(u_i) \tag{15}$$

Por fim a camada de saída realiza a classificação por meio da técnica denominada função de Base Radial Euclidiana (*RBF*, do inglês *Euclidean Radial Bases Function*) o calculo é executado mensurando a distancia entre a entrada da camada e cada possível rotulo de classificação. Ou seja, uma vez para cada uma das 84 opções de saída.

4.6.4.2 AlexNet

A arquitetura AlexNet foi proposta em 2012 como uma adaptação da LeNet-5 desenvolvida por Krizhevsky, Sutskever e Hinton. Sua implementação traz como novidade a função de ativação conhecida Unidade Linear Retificada (Relu) e uma técnica conhecida como desligamento (*Dropout*) para reduzir o impacto do excesso de adaptação da rede ao conjunto de dados de treinamento. A técnica de desligamento desativa neurônios com um baixo potencial de ativação para a determinada função, compactando o tamanho da rede e prevenindo que o excesso de conexões ocasione a adaptação excessiva ao conjunto de treinamento. A primeira aplicação desta arquitetura presente na literatura se refere à classificação de aproximadamente um milhão e duzentas mil imagens do conjunto de treinamento LSVRC-2012 em mil classes diferentes. As imagens apresentavam grande variação de tamanho e, portanto, os autores redimensionaram as imagens unificando o tamanho dos vetores de entrada da rede para 256x256 e obtiveram uma taxa de erro de 18,9% consagrando a arquitetura como vencedora do desafio de classificação de imagens em larga escala deste ano (KRIZHEVSKY; SUTSKEVER; HINTON, 2012).

A arquitetura AlexNet contém oito camadas com pesos, os cinco primeiros são convolucionais e os três subsequentes são camadas totalmente conectadas.

A saída da última camada totalmente conectada é alimentada para um softmax de 1.000 rótulos devidamente distribuídos entre as 1.000 classes alvos. A arquitetura maximiza o objetivo da função de regressão logística. Ou seja, maximiza a média probabilística entre os casos de treinamento do rótulo correto sob a distribuição da predição.

Os núcleos da segunda, quarta e quinta camadas convolucionais são conectados apenas àqueles grãos mapas na camada anterior que residem no mesmo GPU. Os núcleos da terceira camada convolucional estão conectados a todos os mapas do núcleo na segunda camada. Os neurônios nas camadas totalmente conectadas estão interligados a todos os neurônios da camada anterior. Camadas de normalização de resposta seguem a as duas primeiras camadas convolucionais. As Camadas de agrupamento baseadas em *maxpooling*. Ambas as camadas de normalização de resposta bem como a quinta camada convolucional. Por meio da função de ativação ReLU a não linearidade é aplicada à saída de cada camada, seja ela convolucional ou totalmente conectada.

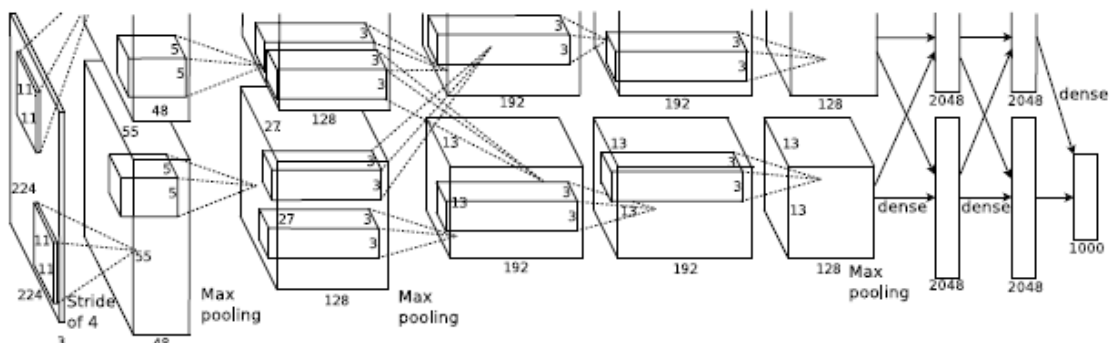
A primeira camada convolucional filtra a imagem de entrada disposta pela matriz $224 \times 224 \times 3$ com 96 núcleos de tamanho $11 \times 11 \times 3$ com passos de quatro pixels, valor correspondente á distância entre os centros de campo receptivos dos neurônios vizinhos em um mapa do núcleo.

A segunda camada convolucional como entrada a resposta normalizada e agrupada correspondente á saída da primeira camada e aplica um filtro com 256 núcleos de tamanho $5 \times 5 \times 48$. As camadas de número três, quatro e cinco, também convolucionais, estão conectadas umas às outras sem qualquer intervenção de funções de agrupamento ou normalização.

A terceira camada é composta por 384 núcleos de tamanho $3 \times 3 \times 256$ ligados às saídas normalizadas e agrupadas da segunda camada convolucional.

A quarta camada tem 384 núcleos de tamanho $3 \times 3 \times 192$ e a quinta camada convolucional tem 256 núcleos de tamanho $3 \times 3 \times 192$. Por fim, as camadas totalmente conectadas são compostas 4096 neurônios cada. A figura 13 ilustra a arquitetura descrita:

Figura 13 –Arquitetura AlexNet



Fonte: (KRIZHEVSKY; SUTSKEVER; HINTON, 2012)

4.6.4.3 VGGNet

A arquitetura VGGNet foi aplicada em 2014 por Simonyan e Zisserman durante a competição ILSVRC 2014. Sua arquitetura é composta por vinte e uma camadas agrupadas em cinco blocos de convoluções seguidas por camadas de agrupamento e um sexto bloco composto por camadas totalmente conectadas (SIMONYAN; ZISSERMAN, 2014).

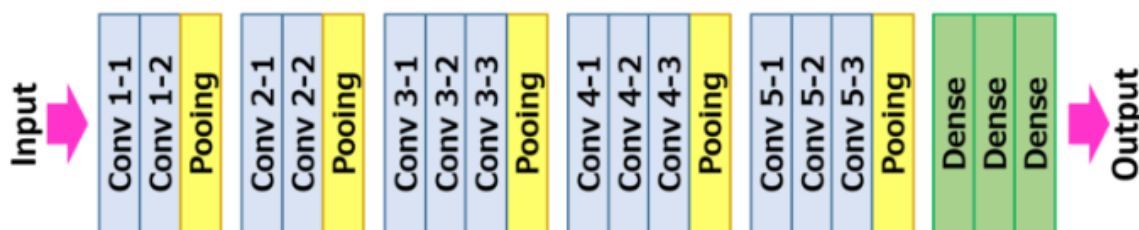
O problema abordado pela arquitetura VGG foi o mesmo para o qual foi proposta a AlexNet, ou seja, a classificação de aproximadamente um milhão e duzentas mil imagens do conjunto de treinamento LSVRC-2012 em mil classes diferentes. Porém esta arquitetura foi proposta apenas dois anos depois e se consagrou com o segundo melhor desempenho do ano de 2014. Sua taxa de erro foi de apenas 7.3% contra 18,9% obtido anteriormente pela arquitetura AlexNet.

A entrada da rede é composta por imagens no padrão de cores *RGB* (vermelho, verde e azul) redimensionadas para matrizes de 224×224 pixels de tamanho fixo. O único pré-processamento realizado foi a subtração do valor *RGB* médio de cada pixel computado no conjunto de treinamento. A imagem então permeia uma trilha de camadas convolucionais nas quais são aplicados filtros de recursos de proporção 3×3 e 1×1 (SIMONYAN; ZISSERMAN, 2014).

A aplicação da convolução é fixa a 1 pixel. O preenchimento espacial da entrada das camadas convolucionais é preservado mesmo após a operação de convolução. Em outras palavras, o preenchimento é de 1 pixel para o filtro de 3×3 das camadas. O agrupamento espacial é realizado por cinco camadas de agrupamento *maxpooling* que encerram os blocos de camadas convolucionais (SIMONYAN; ZISSERMAN, 2014).

O agrupamento de recursos é executado em uma janela de 2×2 . A arquitetura apresenta ainda três camadas totalmente conectadas: as duas primeiras contendo 4096 parâmetros cada. A terceira realiza a classificação final proposta pelo desafio ILSVRC e, portanto, contém 1.000 parâmetros, respectivamente um para cada classe alvo. A camada final é a camada *softmax*. A configuração das camadas totalmente conectadas é a mesma em todas as redes. Todas as camadas ocultas utilizam a função de ativação ReLU garantindo a não linearidade, o aprendizado por sua vez deriva da função de Backpropagation com momentum. A figura 14 ilustra a arquitetura descrita:

Figura 14 – Arquitetura VGG



Fonte: (SIMONYAN; ZISSERMAN, 2014)

4.6.4.4 GoogLeNet

Assim como a arquitetura VGG-16, a arquitetura GoogLeNet foi apresentada na competição ILSVRC de 2014 e se tornou a campeã do desafio obtendo um erro de apenas 6,67%. Esta arquitetura é composta por 22 camadas e tem grande destaque em seu desempenho em função da redução do número de parâmetros de 60 milhões (AlexNet) para apenas 4 milhões (SZGEDY et al., 2015).

Segundo Szegedy e colaboradores (2015), a ideia principal da arquitetura GoogLeNet é obter uma estrutura local esparsa ideal para uma rede de visão convolucional poder ser aproximada e ao mesmo tempo conter uma grande densidade de camadas (SZGEDY et al., 2015).

A arquitetura da rede é construída a partir de blocos convolucionais denominados *Inceptions Modules*. A estratégia dos autores para alcançar o objetivo proposto foi encontrar a construção ideal para um bloco e repeti-la espacialmente. Portanto, a arquitetura foi projetada camada por camada analisando as estatísticas de correlação da última camada, e inserindo em grupos de unidades com alta correlação. Esses blocos formam as unidades da próxima camada e então são conectados as unidades na camada anterior. Cada unidade de uma camada anterior corresponde a alguma região da imagem de entrada e essas unidades são agrupadas em bancos de filtros.

Nas camadas mais próximas das entradas, unidades correlacionadas se concentrariam nas regiões locais. Assim, a rede obteria muitos blocos de camadas concentrados em uma única região da imagem. No entanto, estes locais poderiam ser cobertos por camadas convolucionais de 1x1 na próxima camada.

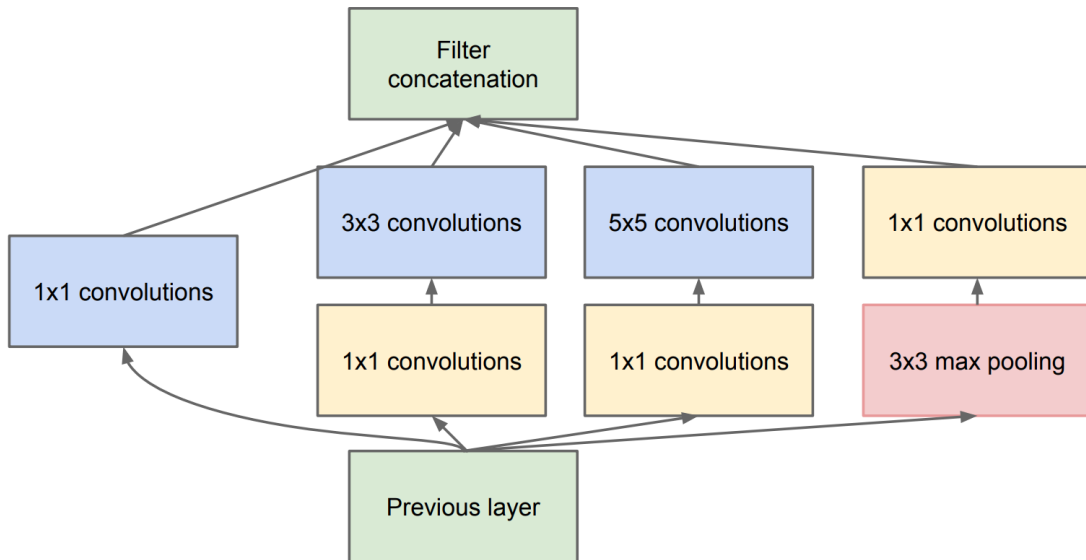
Pode-se esperar um número menor de blocos mais espacialmente espalhados que podem ser cobertos por convoluções ao longo de maiores regiões, e haverá um número decrescente de interligações em regiões maiores.

A fim de evitar problemas de alinhamento de interligações, as arquiteturas anteriores estavam restritas aos tamanhos de filtro 1×1 , 3×3 e 5×5 . Esta decisão foi baseada mais em conveniência do que necessidade. Isso também significa que a arquitetura sugerida é uma combinação de todas essas camadas com seu filtro de saída moldado por bancos concatenados em um único vetor de saída que forma a entrada do próximo estágio. Além disso, uma vez que as operações de agrupamento têm sido essenciais para o sucesso das redes convolucionais atuais, a arquitetura propõe que a adição de um caminho de agrupamento paralelo alternativo em cada estágio deve ter efeito benéfico, também para a acurácia da rede (SZGEDY et al., 2015).

Como os *Inception modules* são empilhados uns nos outros, suas estatísticas de correlação de saída são obrigadas a variar e as características de maior abstração são capturadas pelas camadas mais altas, espera-se que sua concentração espacial diminua. Este fato sugere que a proporção de convoluções 3×3 e 5×5 aumente à medida que avança para as camadas subsequentes (SZGEDY et al., 2015).

Um problema com a inserção dos módulos é que mesmo um número modesto de convoluções 5×5 pode ser proibitivamente caro computacionalmente quando posicionado acima de uma camada convolucional com um grande número de filtros (SZGEDY et al., 2015). Esse problema se torna ainda mais eminente quando as unidades de agrupamento são adicionadas ao módulo. O número de filtros de saída é igual ao número de filtros no estágio anterior. A fusão da saída da camada de agrupamento com saídas das camadas convolucionais levaria a um aumento inevitável no número de saídas de etapa para etapa. Enquanto esta arquitetura pode cobrir a estrutura esparsa ideal, faria isso de forma muito ineficiente, levando a um custo computacional impraticável dentro de alguns estágios. Isso leva à segunda ideia da arquitetura, reduzir criteriosamente a dimensão sempre que os requisitos computacionais aumentarem demais. Este conceito se baseia no sucesso das incorporações, até mesmo os componentes de baixa dimensão podem conter muitas informações sobre uma parte da imagem relativamente grande. No entanto, as incorporações representam informações em uma forma compacta e densa e a informação comprimida é mais difícil de processar. A representação deve ser mantida esparsa na maioria dos lugares pelas condições e comprimir apenas os sinais sempre que eles precisam ser agregados em massa. Isso é, Convoluções 1×1 são usadas para calcular reduções antes das convoluções de maior custo respectivamente 3×3 e 5×5 . Além da redução, as incorporações propostas incluem o uso de funções de ativação linear retificada, tornando-as de dupla finalidade. O resultado final é representado na Figura 15.

Figura 15 –Inception Module



Fonte: (SZGEDY et al., 2015)

Em geral, uma rede Inception é uma rede que consiste em módulos como o exemplo apresentado na imagem empilhados uns sobre os outros, com camadas de agrupamento ocasionais com filtros aplicados em 2 para reduzir pela metade a resolução da rede. Por razões técnicas como memória e eficiência durante o treinamento, a decisão dos autores foi usar módulos somente em camadas mais ao final da rede, as camadas iniciais operam conforme as redes convolucionais tradicionais.

As camadas e parâmetros da arquitetura GoogLeNet são apresentados a Tabela 5.

Tabela 5 - Disposição das camadas e parâmetros da arquitetura GoogLeNet

Tipo da Camada	Tamanho de Saída	Profundidade	#1x1	#3x3 redução	#3x3	#5x5 redução	#5x5	Parâmetros
Convolution	112x112x64	1						2.7K
Max Pool	56x56x64	0						
Convolution	56x56x192	2		64	192			112K
Max Pool	28x28x192	0						
Inception (3a)	28x28x256	2	64	96	128	16	32	159K
Inception (3b)	28x28x480	2	128	128	192	32	96	380K
Max Pool	14x14x480	0						
Inception (4a)	14x14x512	2	192	96	208	16	48	364K
Inception (4b)	14x14x512	2	160	112	224	24	64	437K
Inception (4c)	14x14x512	2	128	128	256	24	64	463K
Inception (4d)	14x14x528	2	112	144	288	32	64	580K
Inception (4e)	14x14x832	2	256	160	320	32	128	840K
Max Pool	7x7x832	0						
Inception (5a)	7x7x832	2	256	160	320	32	128	1072K
Inception (5b)	7x7x1024	2	384	192	384	48	128	1388K
Avg Pool	1x1x1024	0						
Dropout (40%)	1x1x1024	0						
Linear	1x1x1000	1						1000K
Softmax	1x1x1000	0						

Fonte: (SZGEDY et al., 2015)

4.6.4.5 Resnet

A arquitetura ResNet foi desenvolvida por He e colaboradores (2016). Vencedora do ILSRVC 2015, a arquitetura composta por 152 camadas obteve um desempenho sem precedentes atingindo um erro de apenas 3.6% (HE et al., 2016)

Para o treinamento da rede, as imagens do conjunto de dados foram redimensionadas para a escala de 224×224 pixels; A coloração dos pixels foi ajustada pela subtração do valor *RGB* médio de cada pixel computado no conjunto de treinamento; A taxa de aprendizado foi inicializada a partir do valor de 0,1 e é dividida por 10 quando os platôs de erro e os modelos são treinados para até 60×10^4 iterações. O treinamento aplica um decaimento de peso de 0,0001 e um momento de 0,9. As camadas convolucionais utilizam a função de ativação Relu, e as camadas totalmente conectadas a função softmax. A distribuição completa da arquitetura é ilustrada pela Tabela 6.

Tabela 6 - Distribuição completa da arquitetura ResNet

Nome da camada	Tamanho de saída	18-Camada	34-Camada	50-Camada	101-Camada	152-Camada
Conv1	112x112	7x7, 64, Stride 2				
Conv2.x	56x56	3x3 Max Pool, Stride 2				
		$\begin{bmatrix} 3x3, & 64 \\ 3x3, & 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3x3, & 64 \\ 3x3, & 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 64 \\ 3x3, & 64 \\ 1x1, & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 64 \\ 3x3, & 64 \\ 1x1, & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 64 \\ 3x3, & 64 \\ 1x1, & 256 \end{bmatrix} \times 3$
Conv3.x	28x28	$\begin{bmatrix} 3x3, & 128 \\ 3x3, & 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3x3, & 128 \\ 3x3, & 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1x1, & 128 \\ 3x3, & 128 \\ 1x1, & 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1x1, & 128 \\ 3x3, & 128 \\ 1x1, & 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1x1, & 128 \\ 3x3, & 128 \\ 1x1, & 512 \end{bmatrix} \times 8$
Conv4.x	14x14	$\begin{bmatrix} 3x3, & 256 \\ 3x3, & 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3x3, & 256 \\ 3x3, & 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1x1, & 256 \\ 3x3, & 256 \\ 1x1, & 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1x1, & 256 \\ 3x3, & 256 \\ 1x1, & 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1x1, & 256 \\ 3x3, & 256 \\ 1x1, & 1024 \end{bmatrix} \times 36$
Conv5.x	7x7	$\begin{bmatrix} 3x3, & 512 \\ 3x3, & 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3x3, & 512 \\ 3x3, & 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 512 \\ 3x3, & 512 \\ 1x1, & 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 512 \\ 3x3, & 512 \\ 1x1, & 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1x1, & 512 \\ 3x3, & 512 \\ 1x1, & 2048 \end{bmatrix} \times 3$
	1x1	Average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

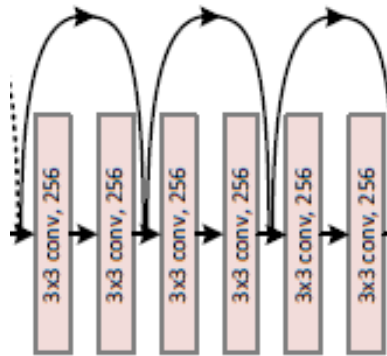
Fonte: (HE et al., 2015)

Apesar do grande número de camadas a verdadeira inovação desta arquitetura está no conceito de aprendizado residual. A precisão das redes tende a diminuir em cenários onde a arquitetura se torna muito densa. Ou seja, apresenta um volume muito grande de camadas. Essa perda de precisão ocorre em função da redução dos gradientes que vão ficando cada vez menores à medida que o sinal é transmitido entre as camadas.

A solução proposta por He e colaboradores foi a introdução de conexões entre a saída de camadas anteriores e a saída de camadas subsequentes. Em outras palavras, em um paradigma de aprendizado residual, a saída de uma camada não é transmitida apenas para as

camadas subsequentes, mas também se propaga para saídas das camadas posteriores. Denotando cada camada por $f(x)$. Em uma rede padrão a saída da segunda camada seria $y = f(x)$ no entanto, em uma rede residual seria $Y = f(x) + x$ (HE et al., 2016). Este efeito é ilustrado na figura 16:

Figura 16 – Aprendizado Residual



Fonte: (HE et al., 2015)

5 METODOLOGIA EXPERIMENTAL

A metodologia experimental utilizada nesta pesquisa foi dividida em cinco tópicos: A coleta de dados, onde se descreve o processo de obtenção das imagens e avaliação dos especialistas; A rotulação das classes, que explica o modo como o estudo constituiu as dezessete classes para uso na modelagem preditiva; Ferramentas utilizadas onde se descreve individualmente cada ferramenta e como foram aplicadas ao longo do projeto; Os cenários de teste que percorrem toda a etapa de preparação e treinamento dos algoritmos. Por fim, o capítulo descreve como serão apresentados os resultados no capítulo 6.

5.1 Coleta de Dados

Este projeto tem a disposição um conjunto de dados composto por 140.446 imagens correspondentes a 57.568 lesões dermatológicas. As imagens foram obtidas entre julho de 2017 e agosto de 2018 por meio de um projeto assistencial filantrópico fruto de uma parceria entre a Sociedade Beneficente Israelita Brasileira Albert Einstein e a prefeitura do município de São Paulo. As lesões foram diagnosticadas por treze especialistas e classificadas em 210 CIDs (Código Internacional de Doenças) diferentes.

A coleta destas imagens foi realizada por técnicos em enfermagem equipados com *smartphone* Samsung Galaxy S7 em cinco pontos da cidade de São Paulo. Os pacientes eram moradores da cidade que aguardavam por atendimento especializado no Sistema Único de Saúde (SUS). Para os casos onde a lesão do paciente careceu de uma biópsia para confirmação da hipótese diagnóstica foi considerado o laudo emitido após o exame de confirmação do diagnóstico; para os demais casos a lesão foi encaminhada para a avaliação de um segundo especialista. Havendo consenso entre as opiniões, o diagnóstico foi considerado correto, caso contrário, divergência de opinião entre os especialistas, um terceiro profissional foi consultado para auxiliar na classificação da lesão.

Destaca-se que como o objetivo do projeto foi avaliar a viabilidade da aplicação das redes neurais profundas para realizar a triagem de lesões dermatológicas por meio de imagens coletadas com base em protocolos e equipamentos disponíveis no Sistema Único de Saúde (SUS). O protocolo de coleta se baseia na captura de três imagens da lesão, uma primeira a 30 centímetros de distância, a segunda a 10 centímetros de distancia e a terceira imagem retirada por um ângulo horizontal para permitir que o dermatologista possa visualizar o relevo da lesão. Todas as fotos foram capturadas por uma câmera de *smartphone* tradicional. A qualidade das imagens presentes no conjunto de dados se mostrou bastante inferior quando

comparada com os experimentos que apresentaram os melhores resultados da literatura. O conjunto de dados apresentou grandes variações nas condições de iluminação, presença de objetos alheios à lesão nas imagens, variação de distância entre o dispositivo e a lesão ao longo das diferentes coletas, ausência de padronização de plano de fundo, mau enquadramento da câmera no momento da aquisição da imagem e falta de foco na lesão conforme apresentado na Figura 3.

Para mitigar o problema de falta de foco nas lesões, todas as imagens foram recortadas individualmente pela ferramenta *Microsoft Paint* de modo que somente a região de interesse relacionada à lesão fosse preservada. Após o recorte, para padronizar a entrada da rede, as imagens foram redimensionadas para o tamanho de 224 x 224 pixels.

5.2 Rotulação das imagens e definição de classes alvo

Com as imagens coletadas foi necessário realizar um estudo para entender a representatividade, a predominância e as características dos tipos de lesões presentes no conjunto de dados.

A rotulação das lesões de pele por parte dos dermatologistas foi realizada por meio do Código Internacional de Doenças (CID-10). Inicialmente, esperava-se que com este trabalho fosse possível criar um classificador capaz de identificar o CID de cada lesão. Entretanto, a partir de uma extração dos laudos, identificou-se que além do elevado número de classes, alguns CIDs possuíam um número bem restrito de exemplares, em torno de duas ou três imagens, tornando inviável o treinamento de um algoritmo para sua identificação. Diante deste desafio, o escopo do projeto foi ajustado para a predição do encaminhamento do paciente, sugestão de prioridade do atendimento e identificação do grupo da lesão do paciente para apoio ao diagnóstico. Dessa maneira, o comitê de dermatologistas participantes do projeto agrupou os 210 CIDs identificados em 17 diferentes grupos, baseando-se na natureza da lesão.

Deste modo as categorias de lesões descritas na Tabela 7 se tornaram os alvos de predição deste estudo. Os CIDs que compõem cada grupo estão presentes no anexo 1 deste documento. Na Tabela 7 ainda se apresentam a categoria da lesão, o encaminhamento feito para o diagnóstico e, por fim, o tipo de prioridade necessário para cada lesão.

Tabela 7 – Classes Alvo

Identificação	Categoria	Encaminhamento	Prioridade
1	Causa Genética	Tele dermatologia	Baixa
2	Causa Metabólica	Dermatologia	Baixa
3	Causa Traumática	Tele dermatologia	Média
4	Cisto Benigno	Dermatologia	Baixa
5	Colagenose	Dermatologia	Alta
6	Distúrbio de Pigmentação	Tele dermatologia	Baixa
7	Distúrbio Psiquiátrico	Tele dermatologia	Média
8	Eczema	Tele dermatologia	Média
9	Infecção Profunda	Dermatologia	Alta
10	Infecção Superficial	Tele dermatologia	Média
11	Inflamação de Causa Desconhecida	Tele dermatologia	Média
12	Lesão Não Agrupada	Tele dermatologia	Baixa
13	Lesão Bolhosa	Tele dermatologia	Alta
14	Reação Adversa a Droga	Tele dermatologia	Média
15	Tumor Benigno	Tele dermatologia	Baixa
16	Tumor Maligno	Biopsia	Alta
17	Tumor Pré-Maligno	Biopsia	Alta

Fonte: Criada pelo autor

5.3 Ferramentas utilizadas

As ferramentas utilizadas para execução deste projeto e os motivos pelo qual foram selecionadas são descritos nesta subseção. Como linguagem de programação utilizou-se Python, em razão da disponibilidade de uma extensa biblioteca de ferramentas de análise de dados de código aberto, *frameworks web* e instrumentos de teste, oferecendo um ecossistema convidativo a programação, principalmente no contexto de redes neurais convolucionais, processamento de imagens e ciência de dados. Para os experimentos de classificação utilizou-se o Keras e Tensorflow. O Keras foi projetado para ser uma *API* (do inglês *Application Programming Interface*) aderente as melhores práticas da aplicação de inteligência artificial e reduzir a carga cognitiva necessária para sua utilização. Adicionalmente oferece APIs consistentes e simples, minimizando o número de ações necessárias para casos de uso comuns e fornecendo *feedbacks* claros aos usuários em caso de erros durante sua execução. O TensorFlow é um sistema de aprendizagem de máquina escalável, de código aberto e capaz de operar em ambientes heterogêneos. A proposta do TensorFlow é facilitar a aplicação dos diversos algoritmos de inteligência artificial que compõe suas bibliotecas provendo aos seus usuários uma abstração simples para utilização dos algoritmos baseada no fluxo dos dados durante o processo de treinamento (ABADI et al., 2016).

5.4 Experimentos

Neste trabalho foram avaliadas algumas das principais arquiteturas de redes neurais profundas, os benefícios da utilização de diferentes paradigmas de transferência de aprendizado e o impacto da variação de parâmetros na acurácia da rede neural artificial. As redes neurais foram aplicadas à classificação de lesões de pele com o objetivo de determinar o encaminhamento correto para o tratamento do paciente, a prioridade do tratamento e fornecer ao médico uma sugestão diagnóstica. A hipótese inicial é de que as redes neurais profundas sejam capazes de realizar essa classificação de forma independente, apenas utilizando as imagens e seus respectivos rótulos em uma abordagem de aprendizado supervisionado sem a necessidade da aplicação de outras técnicas para segmentar as imagens ou auxiliar na extração de características. O estudo realizou dois experimentos:

5.4.1 Primeiro Experimento

O primeiro experimento utilizou 1.000 imagens do conjunto de dados do processo de aquisição. Com o intuito de evitar que o conjunto de dados utilizado ficasse desbalanceado em vista da disparidade da quantidade de imagens em cada grupo, foi utilizado o mesmo número de imagens em cada um dos cinco grupos para treinamento e validação. As imagens utilizadas pertencem a cinco grupos com maior volume de diagnósticos (200 imagens de cada grupo). Os cinco grupos utilizados no experimento totalizam 65% de todo o volume de diagnósticos presentes nos 17 grupos. Para avaliar o desempenho da rede 200 imagens foram utilizadas para validação (40 de cada grupo). A escolha das imagens foi aleatória, o único viés de seleção foi o grupo ao qual a lesão pertence. Para o treinamento foi utilizada a taxa de momentum com valor de 0.9 conforme a configuração original das arquiteturas VGG, ResNet e GoogLeNet. Para a Taxa de Aprendizado (TA), foram utilizados os valores referentes ao melhor desempenho de cada arquitetura aplicado ao contexto de classificação de lesões dermatológicas na literatura.

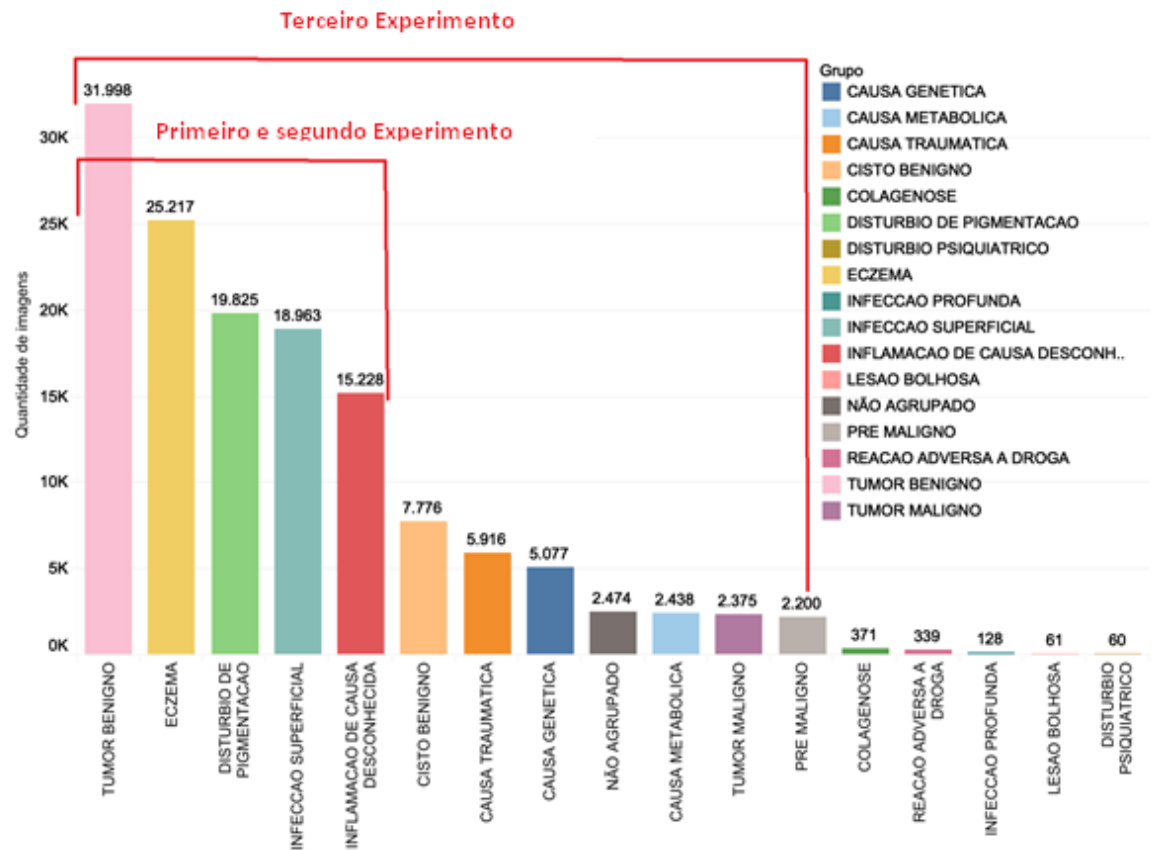
Para a arquitetura ResNet, a TA foi inicializada a 10^{-1} e seu valor foi dividido por 10 a cada 60×10^4 interações com o conjunto de treinamento (HAN et al, 2018). Nos cenários com a arquitetura VGG a taxa de aprendizado foi inicializada com o valor de 10^{-2} e dividida por 10 sempre que Acurácia do conjunto de validação era estabilizada ao longo das épocas do treinamento (MAHBOD; ECKER; ELLINGER, 2017). A arquitetura GoogLenet foi

inicializada com a Taxa de Aprendizado de 10^{-2} com um decréscimo de 4% em seu valor a cada 8 épocas (ZHANG et al, 2017).

As imagens do conjunto de dados foram processadas em grupos de 32 imagens simultaneamente. O treinamento foi executado ao longo de 500 épocas.

Os grupos selecionados para este primeiro experimento e a distribuição das lesões em cada grupo são apresentados na Figura 17.

Figura 17 – Distribuição de lesões por grupo



Fonte: Criada pelo autor

Os grupos selecionados para este primeiro experimento e a distribuição das lesões em cada grupo são apresentados na Figura 17.

Exemplos das imagens de cada grupo são apresentados na Figura 18, da esquerda para a direita respectivamente: cisto benigno, eczema, tumor benigno, infecção superficial e distúrbio de pigmentação.

Figura 18 - Imagens de exemplo das classes dos experimentos 1 e 2

Fonte: Criada pelo autor

Nesse experimento foram avaliados 9 cenários que constitui de variações de arquitetura e estratégia de inicialização de pesos das redes neurais profundas. Foram avaliadas as arquiteturas VGG, GoogLeNet e ResNet com abordagens de inicialização de pesos fixados como extratores de características (EMEF), extensão do modelo e reinicialização dos pesos (EMRP) e inicialização randômica (IR). Para transferência de aprendizado serão utilizados os pesos provenientes do treinamento do conjunto de dados ImageNet ao qual foi aplicado para treinamento das arquiteturas na competição ILSVRC (SIMONYAN; ZISSERMAN, 2014; SZGEDY et al., 2015; HE et al., 2015). Todos os modelos foram treinados e testados com API do *Keras* usando *Tensorflow* como *backend* em duas instâncias da Amazon: p2.xlarge com uma única GPU Nvidia Tesla k80, g3.4xlarge com uma única GPU Nvidia Tesla M60. Os cenários avaliados no primeiro experimento são apresentados na Tabela 8.

Tabela 8 – Cenários Primeiro Experimento

Cenário	Arquitetura	Estratégia de Inicialização dos Pesos	TA
1	VGG	EMEF	10^{-2}
2	VGG	EMRP	10^{-2}
3	VGG	IR	10^{-2}
4	GoogLeNet	EMEF	10^{-2}
5	GoogLeNet	EMRP	10^{-2}
6	GoogLeNet	IR	10^{-2}
7	ResNet	EMEF	10^{-1}
8	ResNet	EMRP	10^{-1}
9	ResNet	IR	10^{-1}

Fonte: Criada pelo autor

5.4.2 Segundo Experimento

O segundo experimento utilizou as mesmas 1.000 imagens do primeiro experimento. Para o treinamento foi utilizada a taxa de momentum com valor de 0.9. Entretanto, a TA foi fixada em 10^{-6} para todas as arquiteturas. as imagens do conjunto de dados foram processadas em grupos de 32 imagens simultaneamente. O treinamento foi executado ao longo de 500 épocas.

Os grupos selecionados para este primeiro experimento e a distribuição das lesões em cada grupo são apresentados na Figura 17.

Exemplos das imagens de cada grupo são apresentados na Figura 18, da esquerda para a direita respectivamente: cisto benigno, eczema, tumor benigno, infecção superficial e distúrbio de pigmentação.

Nesse experimento foram avaliados 9 cenários que constitui de variações de arquitetura e estratégia de inicialização de pesos das redes neurais profundas. Foram avaliadas as arquiteturas VGG, GoogLeNet e ResNet com abordagens de inicialização de pesos fixados como extratores de características (EMEF), extensão do modelo e reinicialização dos pesos (EMRP) e inicialização randômica (IR). Para transferência de aprendizado serão utilizados os pesos provenientes do treinamento do conjunto de dados ImageNet ao qual foi aplicado para treinamento das arquiteturas na competição ILSVRC (SIMONYAN; ZISSERMAN, 2014; SZGEDY et al., 2015; HE et al., 2015). Todos os modelos foram treinados e testados com API do *Keras* usando *Tensorflow* como *backend* em duas instâncias da Amazon: p2.xlarge com uma única GPU Nvidia Tesla k80, g3.4xlarge com uma única GPU Nvidia Tesla M60. Os cenários avaliados no primeiro experimento são apresentados na Tabela 9.

Tabela 9 – Cenários Segundo Experimento

Cenário	Arquitetura	Estratégia de Inicialização dos Pesos	TA
1	VGG	EMEF	10^{-6}
2	VGG	EMRP	10^{-6}
3	VGG	IR	10^{-6}
4	GoogLeNet	EMEF	10^{-6}
5	GoogLeNet	EMRP	10^{-6}
6	GoogLeNet	IR	10^{-6}
7	ResNet	EMEF	10^{-6}
8	ResNet	EMRP	10^{-6}
9	ResNet	IR	10^{-6}

Fonte: Criada pelo autor

5.4.3 Terceiro Experimento

Após a análise dos resultados iniciais que serão apresentados no Capítulo 6 foi realizado o terceiro experimento, utilizando 24.000 imagens para treinamento, as quais pertencem a doze grupos de lesões distintos como indicado na Figura 14. Os doze grupos selecionados correspondem respectivamente aos grupos com maior incidência de casos totalizando 99,32% do volume total de diagnósticos realizados, desconsiderando apenas os grupos que não alcançaram a marca de 2.000 imagens distintas. A seleção das imagens foi aleatória, o único viés de seleção foi o grupo ao qual a lesão pertencia.

Neste experimento foram avaliados 27 cenários distintos obtidos por meio da variação da arquitetura, da estratégia de inicialização dos pesos e da taxa de aprendizado. Foram avaliadas as arquiteturas VGG, GoogLeNet e ResNet com as estratégias de inicialização de pesos EMEF, EMRP e IR. O conjunto de base aplicado para abordagens de inicialização de pesos também será o ImageNet. Neste experimento o valor da taxa de aprendizado foi ajustado entre os cenários de teste. A variação do parâmetro foi de $1e-03$, $1e-06$ e $1e-09$ respectivamente.

Os demais parâmetros das arquiteturas seguiram a configuração apresentada na literatura (SIMONYAN; ZISSERMAN, 2014; SZGEDY et al., 2015; HE et al., 2015). Para o treinamento foi utilizada a taxa de momentum com valor de 0.9, as imagens do conjunto de

dados foram processadas em grupos de 32 imagens simultaneamente. O treinamento foi executado ao longo de 1.000 épocas.

Para evitar que o conjunto de dados utilizado ficasse desbalanceado em vista da disparidade da quantidade de imagens em cada grupo, foi utilizado o mesmo número de imagens em cada um dos doze grupos para treinamento, validação e teste. Para a validação foram utilizadas 200 imagens aleatórias de cada grupo, as quais não estavam presentes no conjunto de treinamento.

Os grupos selecionados para os cenários deste segundo experimento e a distribuição de lesões em cada grupo podem ser vistos na Figura 17.

Exemplos das imagens de cada grupo são apresentados na Figura 19 correspondendo as seguintes classes da esquerda para a direita e de cima para baixo respectivamente: cisto benigno, eczema, tumor benigno, infecção superficial, causa genética, causa metabólica, causa traumática, inflamação de causa desconhecida, distúrbio de pigmentação, tumor pré-maligno, lesão não agrupada e tumor maligno.

Figura 19 - Imagens de exemplo das classes do experimento 3



Fonte: Criada pelo autor

Todos os modelos de redes neurais foram treinados e testados com API do Keras usando Tensorflow como *backend* em duas instâncias da Amazon: p2.xlarge com uma única GPU Nvidia Tesla k80, g3.4xlarge com uma única GPU Nvidia Tesla M60. Os cenários avaliados no terceiro experimento são apresentados na tabela 10:

Tabela 10 – Cenários Terceiro Experimento

Cenário	Arquitetura	Estratégia de Inicialização dos Pesos	TA
1	VGG	EMEF	10^{-3}
2	VGG	EMRP	10^{-3}
3	VGG	IR	10^{-3}
4	GoogLeNet	EMEF	10^{-3}
5	GoogLeNet	EMRP	10^{-3}
6	GoogLeNet	IR	10^{-3}
7	ResNet	EMEF	10^{-3}
8	ResNet	EMRP	10^{-3}
9	ResNet	IR	10^{-3}
10	VGG	EMEF	10^{-6}
11	VGG	EMRP	10^{-6}
12	VGG	IR	10^{-6}
13	GoogLeNet	EMEF	10^{-6}
14	GoogLeNet.	EMRP	10^{-6}
15	GoogLeNet	IR	10^{-6}
16	ResNet	EMEF	10^{-6}
17	ResNet	EMRP	10^{-6}
18	ResNet	IR	10^{-6}
19	VGG	EMEF	10^{-9}
20	VGG	EMRP	10^{-9}
21	VGG	IR	10^{-9}
22	GoogLeNet	EMEF	10^{-9}
23	GoogLeNet	EMRP	10^{-9}
24	GoogLeNet	IR	10^{-9}
25	ResNet	EMEF	10^{-9}
26	ResNet	EMRP	10^{-9}
27	ResNet	IR	10^{-9}

Fonte: Criada pelo autor

5.5 Apresentação dos Resultados

Os resultados deste trabalho serão apresentados em tabelas constituídas pelas colunas a seguir:

Arquitetura: A coluna Arquitetura se refere à arquitetura da rede neural aplicada no cenário o qual os resultados estão sendo apresentados.

EIP: A coluna EIP introduz a estratégia de inicialização dos pesos utilizada no cenário apresentado.

TA: A coluna TA representa o valor da taxa de aprendizado aplicada à correção do erro obtido durante a classificação das imagens do conjunto de treinamento.

AT: A coluna AT apresenta a acurácia obtida na classificação do conjunto de dados de treinamento. A acurácia se refere ao percentual de acertos obtidos na classificação das imagens de um determinado conjunto.

AV: A coluna AV se refere à acurácia obtida na classificação das imagens do conjunto de dados de validação. O conjunto de dados de validação é, por definição, um conjunto de imagens do mesmo domínio das imagens utilizadas no treinamento, porém as imagens do treinamento não podem estar presentes no conjunto de validação de modo que a rede não seja ajustada para classificar especificamente as imagens utilizadas na validação assim como ocorre no treinamento. Ao avaliar apenas a acurácia obtida no conjunto de dados de treinamento (AT), não é possível verificar a capacidade de generalização da rede e a presença de *overfitting*. Consequentemente, a acurácia do conjunto de validação é um indicador mais realista do desempenho do algoritmo de classificação e será, portanto, o principal indicador de sucesso dos cenários de teste apresentados.

ET: A coluna ET representa o valor do erro obtido na classificação do conjunto de treinamento. O valor do erro foi calculado utilizando a equação do erro quadrático médio.

EV: A coluna EV apresenta o valor do erro obtido na classificação do conjunto de validação. Assim como em ET, o valor do erro foi calculado utilizando a equação do erro quadrático médio.

ES: A coluna ES apresenta a especificidade obtida na classificação do conjunto de validação.

SV: A coluna SV apresenta a sensibilidade obtida na classificação do conjunto de validação.

6 RESULTADOS EXPERIMENTAIS

Este capítulo apresenta os resultados experimentais obtidos a partir da metodologia anteriormente apresentada. Os resultados serão apresentados separados por experimentos.

6.1 Primeiro Experimento

Os resultados obtidos no primeiro experimento são apresentados na Tabela 11.

Tabela 11 – Resultados do Primeiro Experimento

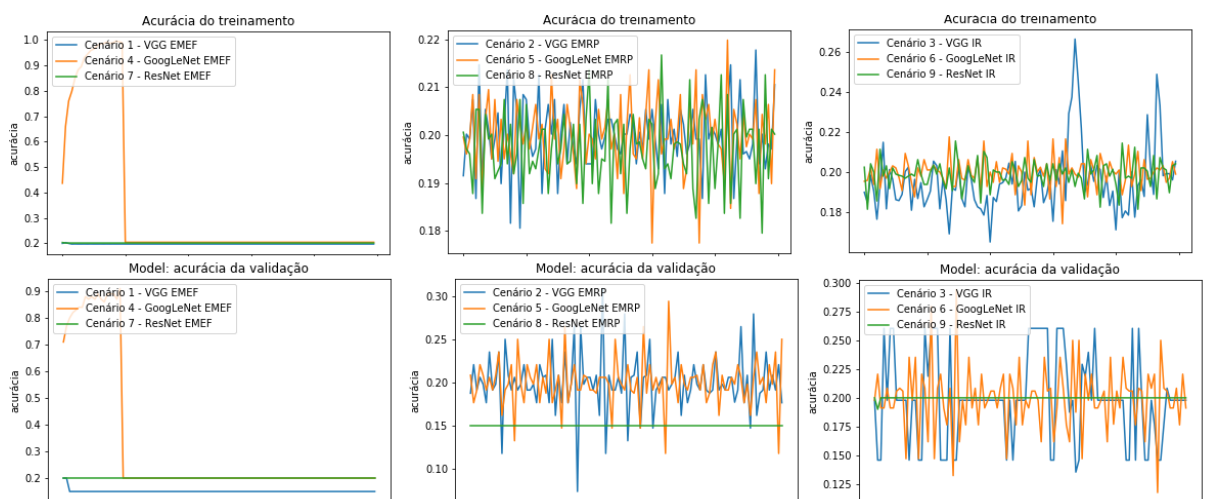
Cenário	Arquitetura	EIP	TA	AT	AV	ET	EV	ESV	SV
1	VGG	EMEF	10^{-2}	20%	20%	1.7511	1.6637	80%	20%
2	VGG	EMRP	10^{-2}	20%	15%	1.6809	1.6434	80%	20%
3	VGG	RI	10^{-2}	18%	26%	1.694	1.6094	80%	20%
4	GoogLeNet	EMEF	10^{-2}	21%	25%	1.7363	1.6406	81,12%	25,2%
5	GoogLeNet	EMRP	10^{-2}	20%	20%	1.6825	1.7704	77,47%	11,20%
6	GoogLeNet	RI	10^{-2}	20%	19%	1.685	2.2818	79,50%	17,60%
7	ResNet	EMEF	10^{-1}	20%	15%	1.6269	0.04	79,75%	19%
8	ResNet	EMRP	10^{-1}	20%	20%	2.1515	1.8448	79,69%	18,66%
9	ResNet	RI	10^{-1}	20%	20%	1.833	2.3286	79,35%	16,79%

Fonte: Criada pelo autor

As curvas de aprendizado das redes neurais ao longo das 500 épocas deste experimento são apresentadas na Figura 20. Os gráficos da linha superior da imagem apresentam a acurácia obtida na classificação do conjunto de treinamento e os gráficos da linha inferior apresentam a acurácia registrada para o conjunto de validação.

O primeiro gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos EMEF. Ou seja, os cenários 1, 4 e 7. O segundo gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos EMRP. Ou seja, os cenários 2, 5 e 8. Por fim, o terceiro gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos IR. Ou seja, os cenários 3, 6 e 9.

Figura 20 – Curva de aprendizado do experimento 1



Fonte: Criada pelo autor

Neste experimento, observou-se que em todos os cenários as arquiteturas não foram capazes de ajustar os pesos de modo à aprender como classificar as imagens de maneira apropriada. Durante todas as épocas ambos os conjuntos de classificação e validação apresentaram a grande maioria de seus elementos classificados em uma mesma classe o que gerou a acurácia aproximada de 20% para todos os cenários. A classificação realizada pelo algoritmo que obteve o melhor desempenho de cada arquitetura no conjunto de validação pode ser observada na Figura 21 sendo respectivamente GoogLeNet com estratégia de

inicialização de pesos com extratores fixos (25%) VGG com estratégia de inicialização de pesos randômica (26%) e ResNet com estratégia de inicialização com extratores fixos:

Figura 21 – Matrizes de confusão do experimento 1

GoogLeNet					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infecção Superficial	Cisto Benigno
Tumor Benigno	0	2	38	0	0
Eczema	0	10	30	0	0
Disturbio de Pigmentação	0	0	40	0	0
Infecção Superficial	0	2	38	0	0
Cisto Benigno	0	2	38	0	0

vgg					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infecção Superficial	Cisto Benigno
Tumor Benigno	0	0	14	0	26
Eczema	0	0	16	8	16
Disturbio de Pigmentação	0	0	22	0	18
Infecção Superficial	0	0	6	14	20
Cisto Benigno	0	0	22	2	16

ResNet					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infecção Superficial	Cisto Benigno
Tumor Benigno	0	0	40	0	0
Eczema	0	0	40	0	0
Disturbio de Pigmentação	0	0	40	0	0
Infecção Superficial	0	0	40	0	0
Cisto Benigno	0	0	40	0	0

Fonte: Criada pelo autor

Com base nestes resultados identificou-se que a Taxa de Aprendizado aplicada nos melhores resultados apresentados na literatura para classificação de imagens adquiridas por meio dos processos e equipamentos de padrão mais elevado, pode não ser a melhor opção para a abordagem proposta de classificação de lesões provenientes das vias de atendimento tradicionais do Sistema Único de Saúde. Esta hipótese embasou o Segundo Experimento onde uma taxa de aprendizado da ordem de 10^{-6} foi aplicada a todas as arquiteturas e estratégias de inicialização de pesos.

6.2 Segundo Experimento

Os resultados obtidos no segundo experimento são apresentados na Tabela 12.

Tabela 12 – Resultados do Segundo Experimento

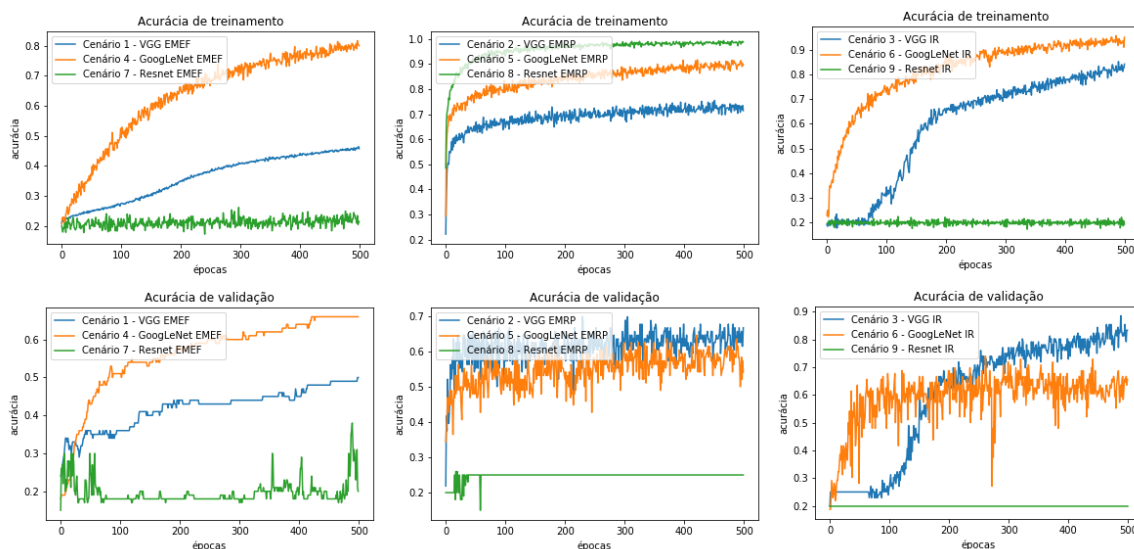
Cenário	Arquitetura	EIP	TA	AT	AV	ET	EV	ESV	SV
1	VGG	EMEF	10 ⁻⁶	45,76%	50%	1.3091	1.3004	88%	57%
2	VGG	EMRP	10 ⁻⁶	71,24%	66,67%	0.7346	1.0217	91%	70%
3	VGG	RI	10⁻⁶	84,17%	83,33%	0.4234	0.5604	95%	82%
4	GoogLeNet	EMEF	10 ⁻⁶	80%	66%	0.5683	0.9638	90%	65%
5	GoogLeNet	EMRP	10 ⁻⁶	88,93%	54,17%	0.2801	2.0313	88%	56%
6	GoogLeNet	RI	10 ⁻⁶	95,06%	65,62%	0.1320	1.3063	91%	65%
7	ResNet	EMEF	10 ⁻⁶	21,50%	20%	0.6022	1.6033	80%	20%
8	ResNet	EMRP	10 ⁻⁶	98,86%	25%	0.039	4.512	80%	20%
9	ResNet	RI	10 ⁻⁶	19,46%	20%	1.6094	1.6094	80%	20%

Fonte: Criada pelo autor

As curvas de aprendizado das redes neurais ao longo das 500 épocas deste experimento são apresentadas na Figura 22. Os gráficos da linha superior da imagem apresentam a acurácia obtida na classificação do conjunto de treinamento e os gráficos da linha inferior apresentam a acurácia registrada para o conjunto de validação.

O primeiro gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos EMEF. Ou seja, os cenários 1, 4 e 7. O segundo gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos EMRP. Ou seja, os cenários 2, 5 e 8. Por fim, o terceiro gráfico de cada linha apresenta a acurácia obtida pelas três arquiteturas quando utilizadas com a estratégia de inicialização dos pesos IR. Ou seja, os cenários 3, 6 e 9.

Figura 22 – Curva de aprendizado do experimento 2



Fonte: Criada pelo autor

Neste experimento, o melhor resultado obtido para acurácia do conjunto de dados de validação foi no cenário 3 composto pela arquitetura VGG com inicialização de pesos randômica (IR). A acurácia do grupo de validação do cenário 3 atingiu de 83%, superando a acurácia de especialistas registrada na literatura de 72% para avaliação presencial e 77% para avaliação por Telemedicina (LEVIN; WARSHAW, 2009). Foi observado que o melhor resultado foi obtido por meio da abordagem de inicialização de pesos randômica (IR), este fato foi atribuído a natureza das imagens ser extremamente específica e significativamente diferente do conjunto de dados ImageNet, utilizado no treinamento do conjunto de base. O número de elementos presentes no conjunto de dados de treinamento foi consideravelmente menor do que o apresentado na literatura onde mais de 1.200.000 imagens foram utilizadas. Os melhores resultados são provenientes da arquitetura VGG, apesar desta arquitetura possuir o desempenho menos expressivo entre as arquiteturas avaliadas no desafio ILSVRC. Uma hipótese para esta arquitetura ter apresentado o melhor resultado pode ser em função do baixo número de elementos do conjunto de treinamento. Isto se deve ao fato de que a VGG ser uma arquitetura com menor número de camadas e essa característica ter contribuído para que os pesos se adaptassem melhor ao baixo volume de elementos no conjunto de treinamento. Portanto, a hipótese de que um maior número de exemplos poderia melhorar a acurácia da rede se fez pertinente, não somente na abordagem de inicialização randômica como nos demais paradigmas de PA. Observou-se ainda uma curva de aprendizado lenta entre as épocas, sobretudo na arquitetura ResNet que praticamente não apresentou melhora no

desempenho ao longo das 500 épocas de treinamento. A arquitetura ResNet é consideravelmente mais densa do que a VGG e a GoogLeNet e mesmo que a proposta da técnica de aprendizado residual seja potencializar os sinais transmitidos entre as camadas da rede, o baixo número de exemplos, somado a uma taxa de aprendizado potencialmente inadequada poderiam ser os motivos para a baixa acurácia apresentada por esta arquitetura. A classificação realizada pelo algoritmo que obteve o melhor desempenho de cada arquitetura no conjunto de validação pode ser observada na Figura 23 sendo a arquitetura GoogLeNet com a estratégia de inicialização de pesos com extratores fixos, a arquitetura ResNet com extensão de modelo e a VGG com a estratégia de inicialização de pesos randômica.

Figura 23 – Matrizes de confusão do experimento 2

GoogLeNet					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infeção Superficial	Cisto Benigno
Tumor Benigno	21	9	3	0	7
Eczema	10	25	5	0	0
Disturbio de Pigmentação	0	2	18	6	14
Infeção Superficial	4	6	3	27	0
Cisto Benigno	2	0	9	8	21

vgg					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infeção Superficial	Cisto Benigno
Tumor Benigno	31	0	4	5	0
Eczema	0	37	0	0	3
Disturbio de Pigmentação	7	1	28	1	3
Infeção Superficial	0	0	0	39	1
Cisto Benigno	0	0	8	0	32

ResNet					
	Tumor Benigno	Eczema	Disturbio de Pigmentação	Infeção Superficial	Cisto Benigno
Tumor Benigno	0	0	32	3	5
Eczema	0	0	29	11	0
Disturbio de Pigmentação	0	0	40	0	0
Infeção Superficial	0	3	33	4	0
Cisto Benigno	0	4	23	7	6

Fonte: Criada pelo autor

6.3 Terceiro Experimento

Os resultados obtidos no Terceiro experimento são apresentados na Tabela 13:

Tabela 13 - Resultados do Terceiro Experimento

Cenário	Arquitetura	EIP	TA	AT	AV	ET	EV	ESV	SV
1	VGG	EMEF	10 ⁻³	99,97%	87%	0.0120	0.7359	96,76%	87,76%
2	VGG	EMRP	10 ⁻³	76,61%	66,66%	0.5899	0.8782	91,50%	68,2%
3	VGG	RI	10 ⁻³	97,72%	86,45%	0.0722	0.7367	96,51%	86,6%
4	GoogLeNet	EMEF	10 ⁻³	100%	88%	0.0002	0.6811	88,59%	97,01%
5	GoogLeNet	EMRP	10 ⁻³	92,46%	58,33%	0.1943	2.027	89,75%	62%
6	GoogLeNet	RI	10 ⁻³	98,09%	90,62%	0.0774	0.4928	97,73%	90,86%
7	ResNet	EMEF	10 ⁻³	19,53%	20%	1.6094	1.6094	80%	20%
8	ResNet	EMRP	10 ⁻³	98,55%	25%	0.0288	8.040	80%	20%
9	ResNet	RI	10 ⁻³	15,27%	14%	1.6094	1.6094	79%	16%
10	VGG	EMEF	10 ⁻⁶	51,92%	60%	1.2061	1.181	89,95%	62,26%
11	VGG	EMRP	10 ⁻⁶	56,85%	51,04%	1.1603	1.2	87,98%	54,33%
12	VGG	RI	10 ⁻⁶	20%	26%	1.6094	1.6093	80%	20%
13	GoogLeNet	EMEF	10 ⁻⁶	88,16%	84%	0.3768	0.4937	95,94%	85%
14	GoogLeNet	EMRP	10 ⁻⁶	66,21%	40,62%	0.899	1.6716	85,13%	43,3%
15	GoogLeNet	RI	10 ⁻⁶	43,28%	38,54%	1.3286	1.6240	84,65%	40,59%
16	ResNet	EMEF	10 ⁻⁶	17,37%	26%	1.6109	1.608	80,26%	21%
17	ResNet	EMRP	10 ⁻⁶	79,77%	25%	0.5516	1.9700	80%	20%
18	ResNet	RI	10 ⁻⁶	34,87%	30%	1.4680	1.6094	82,74%	33,32%
19	VGG	EMEF	10 ⁻⁹	20%	20%	1.7511	1.6637	80%	20%
20	VGG	EMRP	10 ⁻⁹	20,24%	20,83%	1.6809	1.6434	80%	20%
21	VGG	RI	10 ⁻⁹	20,76%	14,58%	1.694	1.6094	80%	20%
22	GoogLeNet	EMEF	10 ⁻⁹	19,81%	25%	1.7363	1.6406	81,12%	25,2%
23	GoogLeNet	EMRP	10 ⁻⁹	18,26%	11,45%	1.6825	1.7704	77,47%	11,20%
24	GoogLeNet	RI	10 ⁻⁹	20,22%	16,66%	1.685	2.2818	79,50%	17,60%
25	ResNet	EMEF	10 ⁻⁹	22,7%	19%	1.6269	0.04	79,75%	19%
26	ResNet	EMRP	10 ⁻⁹	18,85%	14%	2.1515	1.8448	79,69%	18,66%
27	ResNet	RI	10 ⁻⁹	19,42%	15%	1.833	2.3286	79,35%	16,79%

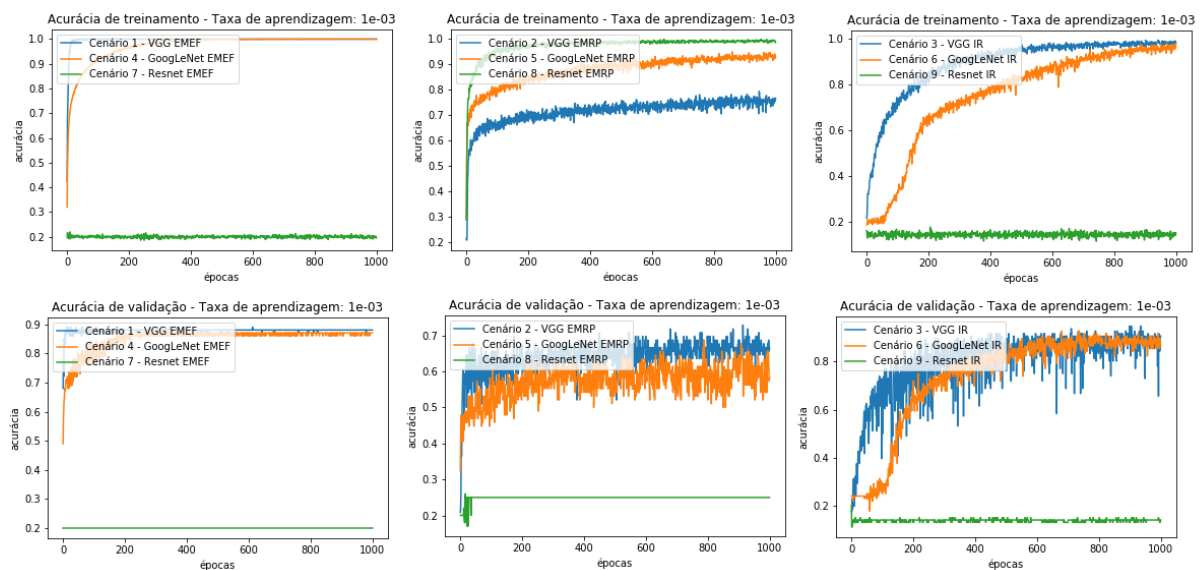
Fonte: Criada pelo autor

As curvas de aprendizado das redes neurais ao longo das 1.000 épocas dos cenários do terceiro experimento são apresentadas nas figuras 24, 25 e 26.

Assim como na Figura 20, as linhas superiores das figuras 24, 25 e 26, apresentam à curva de aprendizado obtida na classificação do conjunto de treinamento e a linha inferior à curva de aprendizado obtida na classificação do conjunto de validação. Apresentando, portanto, o desempenho das três arquiteturas e variando a estratégia de inicialização dos pesos em cada gráfico da esquerda para a direita. Sendo as estratégias EMEF, EMRP e RI respectivamente. Entretanto, como neste experimento além da arquitetura e da estratégia de inicialização, a taxa de aprendizado também foi modificada entre os cenários de teste os resultados foram divididos em três figuras.

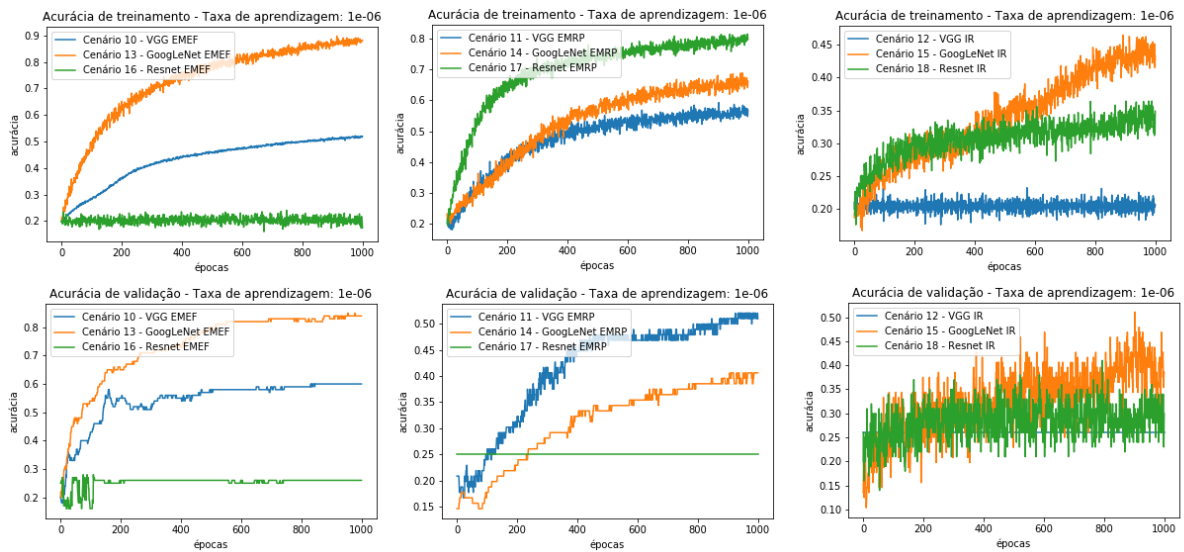
A figura 22 apresenta os resultados dos cenários 1 ao 9 obtidos pela aplicação das três arquiteturas com as três estratégias de inicialização de pesos utilizando a taxa de aprendizado 10^{-3} ; A figura 23 apresenta os resultados dos cenários 10 ao 18 obtidos pela aplicação das três arquiteturas com as três estratégias de inicialização de pesos utilizando a taxa de aprendizado 10^{-6} ; A figura 24 apresenta os resultados dos cenários 19 ao 27 obtidos pela aplicação das três arquiteturas com as três estratégias de inicialização de pesos utilizando a taxa de aprendizado 10^{-9} .

Figura 24 – Curva de aprendizado do experimento 3 com TA 10^{-3}



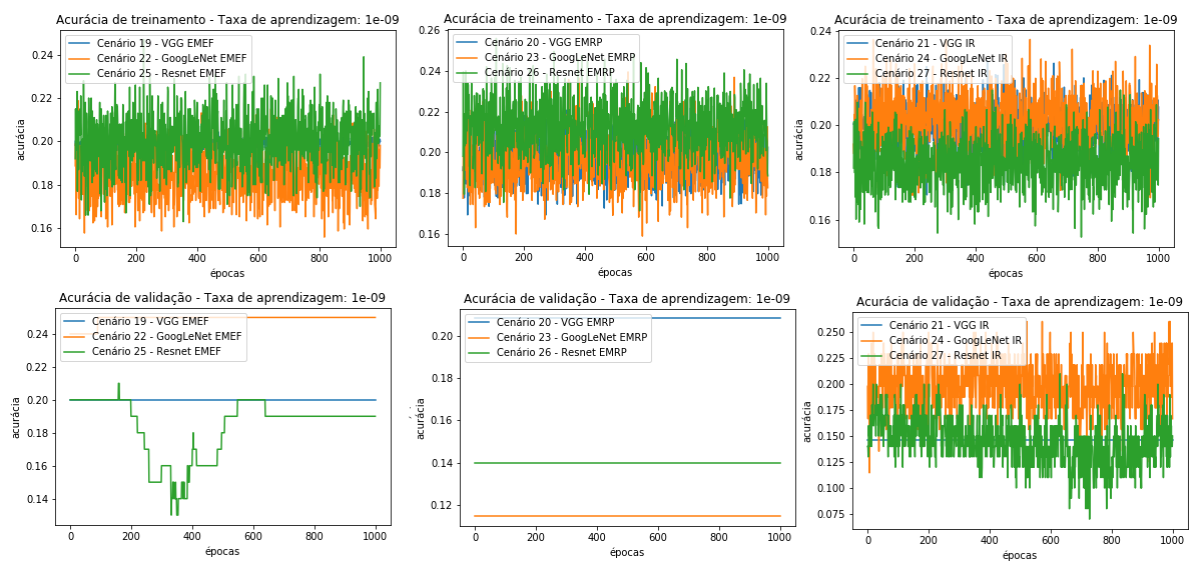
Fonte: Criada pelo autor

Figura 25 – Curva de aprendizado do experimento 3 com TA 10^{-6}



Fonte: Criada pelo autor

Figura 26 – Curva de aprendizado do experimento 3 com TA 10^{-9}



Fonte: Criada pelo autor

O terceiro experimento comprovou as hipóteses formuladas a partir dos resultados do segundo experimento. O aumento do número de imagens no conjunto de dados de treinamento e a utilização de uma taxa de aprendizado com valor intermediário entre os experimentos 1 e 2 proporcionaram uma melhor acurácia em todas as arquiteturas, de modo que as melhores acurácias foram observadas nos cenários com taxa de aprendizado de 10^{-3} . A

arquitetura VGG apresentou resultados equivalentes nos paradigmas de aprendizado RI e EMEF com acurácias de 86,75% e 87% respectivamente. O melhor desempenho deste experimento foi obtido pela arquitetura GoogLeNet que apresentou um comportamento semelhante a VGG em relação aos paradigmas de aprendizado, com uma acurácia ligeiramente melhor para a abordagem RI. A arquitetura apresentou resultados 90,65% de acurácia na abordagem RI e 88% na abordagem EMEF na classificação do conjunto de validação. A arquitetura ResNet atingiu uma acurácia de 98,55% durante o treinamento, entretanto a rede apresentou uma baixa capacidade de generalização obtendo apenas 25% de acurácia no conjunto de dados de validação.

Com o resultado de 90,65% de acurácia no conjunto de validação o cenário 6 foi considerado o cenário com melhor resultado. Para avaliar a consistência dos resultados, um conjunto de teste composto por 5% do total de imagens do estudo, 7.022 imagens pertencentes aos 17 grupos de lesões apresentados na Tabela 7 composto de acordo com o percentual de representatividade da demanda populacional de cada grupo apresentado na Figura 17 foi estruturado. A distribuição das lesões que compõe este conjunto de teste é apresentada na Tabela 14. Os 5 grupos desconsiderados na preparação do conjunto de treinamento estão agrupados no item “Outros Grupos” e foram desconsiderados nesta avaliação, portanto, o conjunto de teste foi composto por 6.975 imagens.

Tabela 14 – Distribuição do Conjunto de Teste

Grupo	Imagens	Percentual	Amostra 5%
Tumor Benigno	31.998	22,78%	1.600
Eczema	25.217	17,95%	1.261
Distúrbio de Pigmentação	19.825	14,12%	992
Infecção Superficial	18.963	13,50%	949
Inflamação de Causa Desconhecida	15.228	10,84%	762
Cisto Benigno	7.776	5,54%	389
Causa Traumática	5.916	4,21%	296
Causa Genética	5.077	3,61%	254
Não Agrupado	2.474	1,76%	123
Causa Metabólica	2.438	1,74%	121
Tumor Maligno	2.375	1,69%	118
Pré-Maligno	2.200	1,57%	110
Outros Grupos	959	0,68%	47
Total de Imagens	140.446	100%	7022

Fonte: Criada pelo autor

Em 12 dos 13 grupos nenhuma das imagens utilizadas no conjunto de testes foi utilizada no conjunto de treinamento e validação. Entretanto, como o grupo Tumor Pré-Maligno possui apenas 2.200 imagens, as quais foram utilizadas nos conjuntos de treinamento e validação, para composição do conjunto de teste deste grupo foram utilizadas as mesmas imagens do conjunto de validação. Ressalta-se que como as imagens do conjunto de validação não influenciam os ajustes da rede, a presença de imagens do conjunto de validação no conjunto de teste não compromete os resultados obtidos.

O conjunto de teste foi classificado pelo algoritmo com melhor desempenho de cada uma das arquiteturas e apresentou uma acurácia de 89,72% para GoogLeNet com estratégia de inicialização de pesos randômica, 81,62% para a arquitetura VGG com estratégia de inicialização de pesos com extratores fixos e 13,52% para a arquitetura ResNet com estratégia de reinicialização de pesos . O Resultado desta classificação pode ser observado nas matrizes de confusão apresentadas nas Figuras 27, 28 e 29 respectivamente.

Figura 27 – Matriz de confusão conjunto de teste - arquitetura GoogLeNet

	Causa Genética	Causa Metabólica	Causa Traumática	Cisto Benigno	Distúrbio de Pigmentação	Eczema	Infecção Superficial	Inflamação de Causa Desconhecida	Não Agrupado	Tumor Benigno	Tumor Maligno	Tumor Pré-Maligno
Causa Genética	236	1		9		53	9		3	38		
Causa Metabólica		119	2			34	7	6	4		4	
Causa Traumática	1		272			3	5		1	14		2
Cisto Benigno	5			352	27	24	23	12	5			
Distúrbio de Pigmentação			4	7	898	36	29	20	5	11	2	
Eczema			2	2		1041			3			
Infecção Superficial			13		29		859		2			
Inflamação de Causa Desconhecida	3			9		2		690				
Não Agrupado	9		3	7	23	68		34	94		3	3
Tumor Benigno		1			11				1	1488		0
Tumor Maligno					4		15		2	20	108	4
Tumor Pré-Maligno				3			2		3	29	1	101

Fonte: Criada pelo autor

Figura 28 – Matriz de confusão conjunto de teste - arquitetura VGG

	Causa Genética	Causa Metabólica	Causa Traumática	Cisto Benigno	Distúrbio de Pigmentação	Eczema	Infecção Superficial	Inflamação de Causa Desconhecida	Não Agrupado	Tumor Benigno	Tumor Maligno	Tumor Pré-Maligno
Causa Genética	217				75							
Causa Metabólica		95	30	14		114	26	6	10		1	
Causa Traumática			240		53		33			23		
Cisto Benigno	8			307				12	21		4	
Distúrbio de Pigmentação		1			821	217	28	56				11
Eczema	10			1		793			22	42	9	
Infecção Superficial				22		59	841		18			2
Inflamação de Causa Desconhecida		18			10	2		654		12	4	
Não Agrupado			17	10	14		7	34	45			4
Tumor Benigno	19	7		35		76	9			1523	3	
Tumor Maligno					8		5		3		96	32
Tumor Pré-Maligno			9		11				4		1	61

Fonte: Criada pelo autor

Figura 29 – Matriz de confusão conjunto de teste - arquitetura ResNet

	Causa Genética	Causa Metabólica	Causa Traumática	Cisto Benigno	Distúrbio de Pigmentação	Eczema	Infecção Superficial	Inflamação de Causa Desconhecida	Não Agrupado	Tumor Benigno	Tumor Maligno	Tumor Pré-Maligno
Causa Genética												
Causa Metabólica	31	56	78	39	298	387	157	76		432		
Causa Traumática	72	22	21	126	267		1	378		77	118	110
Cisto Benigno												
Distúrbio de Pigmentação		1		3	96	217	6			1		
Eczema				9		199						
Infecção Superficial				8			211			1		
Inflamação de Causa Desconhecida								94				
Não Agrupado	151	42	197	212	331	456	574	214	123	946		
Tumor Benigno										143		
Tumor Maligno						2						
Tumor Pré-Maligno												

Fonte: Criada pelo autor

Observando a distribuição de erros na Matriz de confusão da Figura 27, correspondente a arquitetura GoogLeNet, cenário que apresentou o melhor desempenho, é possível notar uma regularidade no número de erros e acertos de cada classe. Exceto, pela classe “Lesão Não Agrupada”, que apresentou não só um menor número de acertos entre as lesões que pertencem a esta classe, como um maior número de lesões de outras classes classificadas com este rótulo. Este comportamento se deve em razão da maior heterogeneidade das imagens que compõem este conjunto de dados. Por se tratar de um conjunto mais genérico dedicado a lesões com menor volume que não caracterizam uma classificação específica, pode-se observar uma menor semelhança entre as imagens do conjunto de dados e em alguns casos, semelhanças com lesões de outras classes.

Os resultados apresentados pela classificação do conjunto de teste permitem avaliar também a acurácia do algoritmo para definição de prioridade e encaminhamento ao local de atendimento apropriado.

Esta análise ocorreu por meio da observação do encaminhamento e da prioridade proposta pelo algoritmo, independente da classe na qual a lesão foi classificada com base na definição de encaminhamento e prioridade de cada classe apresentada na Tabela 7.

Dentre os 12 grupos de lesões avaliados no conjunto de teste, 6 grupos indicam atendimento com baixa prioridade (3.479 imagens), 4 grupos atendimento com prioridade média (3.268 imagens) e 2 grupos atendimentos com alta prioridade (228 imagens). Em relação ao encaminhamento, 8 grupos (6.237 imagens) indicam encaminhamento para tele dermatologia, 2 grupos (510 imagens) para dermatologia presencial e 2 grupos (22 Imagens) para biópsia.

Este teste apresentou uma acurácia de 92,54% para a tarefa de definição de prioridade e 96,03% para o encaminhamento do paciente. A distribuição da indicação de prioridade do atendimento e o encaminhamento sugerido na classificação das imagens são apresentados nas figuras 30 e 31 respectivamente.

Figura 30 – Matriz de confusão definição de prioridade para atendimento

	Baixa	Média	Alta
Baixa	3354	364	12
Média	64	2887	2
Alta	61	17	214

Fonte: Criada pelo autor

Figura 31 – Matriz de confusão de encaminhamentos

	Teledermatologia	Dermatologia	Biópsia
Teledermatologia	6013	36	10
Dermatologia	149	471	4
Biópsia	75	3	214

Fonte: Criada pelo autor

Avaliando a matriz de confusão de definição de prioridade, observa-se que 93,86% das lesões de alta prioridade receberam uma indicação de priorização pertinente do algoritmo, totalizando 214 de um total de 228 imagens. Além de priorizar adequadamente as lesões pertinentes, o algoritmo priorizou incorretamente 78 casos. Portanto, houve sugestão de alta prioridade em 292 casos, dos quais 214 pertinentes, um resultado correspondente 73,23% de especificidade.

A matriz de confusão de encaminhamentos apresentou percentuais de especificidade equivalentes em todos os desfechos possíveis, 96,41% para encaminhamento ao teledermatologista, 92,35% para atendimento com dermatologista presencialmente e 93,86% para encaminhamento a biópsia.

7 CONCLUSÕES E TRABALHOS FUTUROS

Ao longo da apresentação dos resultados, identificou-se que o cenário com melhor desempenho na tarefa de classificação das lesões foi capaz de priorizar atendimentos de alta prioridade corretamente em 92,54% dos casos. Um mecanismo com benefícios promissores quando comparado ao modelo de atendimento atual onde o primeiro paciente atendido é o primeiro a entrar na fila.

O algoritmo foi capaz de indicar o local adequado para atendimento do paciente corretamente em 96,03% dos casos. Atualmente, a ausência de um algoritmo de triagem eficiente faz com que todos os pacientes precisem ser analisados primeiramente por um teledermatologista e caso necessitem de uma biópsia ou de uma avaliação presencial, são encaminhados à fila novamente. A inserção deste algoritmo pode reduzir o tempo de espera do paciente em até 7 dias úteis, tempo que a equipe de teledermatologia possui para avaliar as lesões. Como os casos de alta prioridade não são tratados a distância, 7 dias a menos de espera pode ser representativo para o desfecho dos pacientes, principalmente em casos de tumores malignos. Observa-se um benefício relevante também em eliminar a etapa de análise do teledermatologista quando for indicado outro tipo de atendimento. Além de reduzir a espera do paciente, a automação da etapa de triagem proporciona um ganho de produtividade de aproximadamente 30% da equipe de teledermatologia. Este percentual é correspondente ao número de lesões com indicação de encaminhamento para biópsia ou avaliação presencial que deixarão de ser avaliados por telemedicina desnecessariamente.

Os resultados apresentados indicam que a aplicação de redes neurais artificiais profundas (*Deep Learning*) para triagem, apoio diagnóstico e sugestão de tratamentos, com base em protocolos assistenciais estabelecidos, no contexto de livre demanda populacional do sistema público de saúde se provou efetiva, apresentando grande potencial para redução das filas de espera para atendimento e do volume de encaminhamentos incorretos para os canais de atendimento especializado por parte dos profissionais de atenção primária.

Contudo, a acurácia obtida foi inferior à apresentada nos trabalhos correlatos analisados. Este resultado foi atribuído à heterogeneidade e a baixa qualidade das imagens que compõe o conjunto de dados utilizado no estudo. Entretanto, como o objetivo do trabalho visa validar a aplicação dos algoritmos em cenários de coleta de imagens com protocolos e equipamentos acessíveis em larga escala pelo Sistema Único de Saúde esta concessão de qualidade do conjunto de dados e conseqüentemente da acurácia do algoritmo se fez necessária.

O trabalho permitiu avaliar o desempenho de 42 cenários diferentes da aplicação das redes neurais profundas com diferentes arquiteturas, estratégias de inicialização de pesos e taxas de aprendizado para o contexto de triagem de lesões dermatológicas identificando os algoritmos que apresentaram melhores resultados dentro do contexto proposto. O melhor desempenho foi obtido pela arquitetura GoogLeNet, treinada com 24.000 imagens ao longo de mil épocas utilizando inicialização randômica como estratégia de inicialização dos pesos e taxa de aprendizado de 10^{-3} . Esta arquitetura obteve uma acurácia de 89,72% para sugestão diagnóstica, 92,54% para indicação de prioridade e 96,52% para o encaminhamento apropriado do paciente no conjunto de teste composto por 6.975 imagens.

Como trabalhos futuros, a partir dos resultados obtidos no segundo experimento identifica-se a grande influência do parâmetro Taxa de Aprendizado no desempenho dos algoritmos de classificação. Em uma análise mais aprofundada da configuração deste parâmetro nos trabalhos correlatos que obtiveram os melhores resultados foi possível identificar a aplicação de valores bem distintos aos aplicados nos cenários avaliados.

Considera-se também a possibilidade de avaliar a acurácia dos algoritmos em conjuntos de dados provenientes de outras fontes além do projeto realizado, comparar o desempenho e produtividade dos médicos especialistas realizando laudos com e sem o auxílio do algoritmo para indicar o agrupamento apropriado da lesão. Comparar a assertividade do encaminhamento ao canal de atendimento adequado por parte dos médicos de atenção primária para especialistas com e sem o auxílio do algoritmo.

Avaliar o desempenho dos algoritmos em cenários mais aprofundados, segregando os grupos de doenças com maior especificidade, de modo que a classificação possa agregar maior valor à assistência. O grupo Tumor Maligno, por exemplo, se subdividido em Melanoma, Carinoma Baso Celular e Carnicoma Espino Celular. Permitirá identificar mais rapidamente o Melanoma que é um diagnóstico mais grave mesmo comparado a outros tumores malignos. Outro exemplo seria a segregação do grupo Infecção Superficial, que poderia ser subdividido em grupos menores que identifiquem a causa da infecção (Bacteriana, Fungica, viral e outras). Esta segregação permitirá, inclusive, indicar sugestões de tratamento ao profissional que utilizar o algoritmo.

Durante o processo de aquisição das imagens, diversas informações epidemiológicas dos pacientes, como sexo e idade são registradas, assim como informações de anamnese referentes às lesões como o tempo de evolução, sangramento, existência de prurido entre outras. Considera-se a hipótese de utilizar também essas informações além das imagens das lesões para realizar a classificação. Avalia-se também a aplicação de técnicas de segmentação

de imagem como ampliação da região de interesse, realce das bordas das lesões, remoção de ruídos e aumento do conjunto de dados por meio de técnicas de *agumentation* para verificar se há melhora na acurácia do algoritmo.

Por fim, consideramos avaliar outras arquiteturas de redes neurais profundas propostas ao longo dos últimos anos na literatura, como por exemplo, as arquiteturas ResNeXT, DenseNet e SE-Net.

8 REFERÊNCIAS BIBLIOGRÁFICAS

- ABADI, M.; BARHAM, P.; CHEN, J.; CHEN, Z.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; IRVING, G.; ISARD, M.; KUDLUR, M.; LEVENBERG, J.; MONGA, R.; MOORE, S.; MURRAY, D. G.; STEINER, B.; TUCKER, P.; VASUDEVAN, V.; WARDEN, P.; WICKE, M.; YU, Y.; ZHENG, X.; BRAIN, G. TensorFlow: A system for large-scale machine learning. In: Symposium on Operating Systems Design and Implementation, 12, 2016, Savannah. *Anais...*Savannah: USENIX, 2016. p. 1-18.
- ANDRADE, M. A.; CHACÓM. P.; MERELO J. J.; MORÁN F. Evaluation of secondary structure of proteins from UV circular dichroism spectra using an unsupervised learning neural network. *Protein Engineering*, Granada, v. 6, n. 4, p. 383-390, 1993.
- BASHEER, I. .; HAJMEER, M. X. *Journal of Microbiological Methods*, Manhattan, 2000, v. 43, n. 1, p. 3–31, 1 dez. 2000.
- BHATTACHARYA, A.; YOUNG, A.; WONG, A.; STALLING, S.; WEI, M.; HADLEY, D. Precision diagnosis of melanoma and other skin lesions from digital images. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, v. 2017, p. 220–226, 2017.
- BOTTOU, L. Large-scale machine learning with stochastic gradient descent. In: International Conference on Computational Statistics, 19, 2010, Paris, *Anais...* Paris, COMPSTAT, 2010 p.177-186.
- CAPDEHOURAT, G.; COREZ, A.; BAZZANO, A.; ALONSO, R.; MUSÉ, P. Toward a combined tool to assist dermatologists in melanoma detection from dermoscopic images of pigmented skin lesions. *Pattern Recognition Letters*, v. 32, n. 16, p. 2187–2196, 2011.
- CELEBI, M. E.; KINGRAVI, H. A.; IYATOMI, H.; ASLANDOGAN, Y. A.; STOECKER, W. V.; MOSS, R. H.; MALTERS, J. M.; GRICHNIK, J. M.; MARGHOOB, A. A.; RABINOVITZ, H. S.; MENZIES, S. W. Border detection in dermoscopy images using statistical region merging. *Skin Research and Technology*, v. 14, n. 3, p. 347–353, 2008.
- CLEVERT, D. A.; UNTERHINER, T.; HOCHREITER, S. Fast and accurate deep network learning by exponential linear units. In: International Conference on Learning Representations, 6., 2015. San Diego, *Anais...*San Diego, ICLR 2015.
- CODELLA, N. C. F.; GUTMAN, D.; CELEBI, M. E.; HELBA, B.; MARCHETTI, M. A.; DUSZA, S. W.; KALLOO, A.; LIOPYRIS, K.; MISHRA, N.; KITTLER, H.; HALPERN, A. Skin lesion analysis toward melanoma detection: A challenge at the 2017 International

symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). *Proceedings - International Symposium on Biomedical Imaging*, v. 2018–April, n. Isbi, p. 168–172, 2018.

DA SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas: curso prático. Artliber, 2010.

DAHL, G. E.; SAINATH, T. N.; HINTON, G. E. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: International Conference on Acoustics, Speech and Signal Processing, 38., 2013. Vancouver, *Anais...* Vancouver, IEEE, 2013. p. 8609-8613

DENNIS, J. E.; SCHANABEL, R. B. *Numerical methods for unconstrained optimization and nonlinear equations*, Philadelphia, Editora Siam, 1983.

DUDA, R. O.; HART, P. E. *Pattern Classification and Scene Analysis*. New York: Wiley, feb. 1973.

ELLIOTT, D. L.: A better activation function for artificial neural networks. Technical Research Report, Marryland, 1993. Disponível em: <https://www.researchgate.net/publication/277299531_A_better_Activation_Function_for_Artificial_Neural_Networks> Acesso em 09 jul. 2019

ESTEVA, A.; KUPREL, B.; NOVOA, R. A.; KO, J.; SWETTER, S. M.; BLAU, H. M.; THRUN, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, v. 542, n. 7639, p. 115–118, 25 fev. 2017.

FEDERMAN, D. G. Comparison of dermatologic diagnoses by primary care practitioners and dermatologists a review of the literature. *Archives of Family Medicine*, v. 8, n. 2, p. 170–172, 1999.

FIX, E.; HODGES, J. Discriminatory analysis. nonparametric discrimination: consistency properties. 1951. School Of Aviation Medicine Randolph Field, Texas. 1951. Disponível em: <<https://apps.dtic.mil/dtic/tr/fulltext/u2/a800276.pdf>> Acesso em: 09 jul. 2019.

FRANK ROSENBLATT. The perceptron: A probabilistic model for information storage and organization. *BrainPsychological Review*, v 65, n. 6 , p. 65-386, 1958.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, n.. 55, p. 119–139, 1997.

FRIEDMAN, NIR; GEIGER, DAN; GOLDSZMIDT, MOISES. Bayesian network classifiers. *Machine Learning*, v. 29, p. 131-163, nov. 1997.

GARNAVI, R.; ALDEEN, M.; CELEBI, M. E.; BHUIYAN, A.; DOLIANITIS, C.;

VARIGOS, G. Automatic segmentation of dermoscopy images using histogram thresholding on optimal color. *Machine Learning*, n. January, p. 126–134, 2012.

GLOBAL DIFUSION OF EHEALTH: MAKING UNIVERSAL HEALTH COVERAGE ACHIEVABLE. Report of the third global survey on eHealth. Geneva: World Health Organization; 2016.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*, Cambridge, Editora: MIT Press, 2016.

GOTTIPATI, V. S. K.; GOYETTE, K.; CHAMSEDDINE, A.; CONSIDINE, B. Deep Pepper: Expert iteration based chess agent in the reinforcement learning setting, 2018. Disponível em: <<https://arxiv.org/abs/1806.00683>>. Acesso em: 09 jul. 2019.

GRZESIAK-KOPEĆ, K.; NOWAK, L.; OGORZALEK, M. Automatic diagnosis of melanoid skin lesions using machine learning methods. *Jornal: Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, Springer, Cham, n. 9119, 2015. p. 577–585.

GUTMAN, D.; CODELLA, N. C. F.; CELEBI, E.; HELBA, B.; MARCHETTI, M.; MISHRA, N.; HALPERN, A. Skin Lesion Analysis toward Melanoma Detection: A Challenge at the International Symposium on Biomedical Imaging (ISBI) 2016, hosted by the International Skin Imaging Collaboration (ISIC). 4 maio 2016.

HAN, S. S.; KIM, M. S.; LIM, W.; PARK, G. H.; PARK, I.; CHANG, S. E. Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning Algorithm. *Journal of Investigative Dermatology*, v. 138, n. 7, p. 1529–1538, jul. 2018.

HAN, S. S.; PARK, G. H.; LIM, W.; KIM, M. S.; NA, J. I.; PARK, I.; CHANG, S. E. Deep neural networks show an equivalent and often superior performance to dermatologists in onychomycosis diagnosis: Automatic construction of onychomycosis datasets by region-based convolutional deep neural network. *PLOS ONE*, v. 13, n. 1, p. e0191493, 19 jan. 2018.

HAYKIN, S. *Redes Neurais: Princípios e prática*. 2 ed., Halmiton: Prentice Hall, 2008.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *Conference on Computer Vision and Pattern Recognition*, 6, 2016, Las Vegas. *Anais....Las Vegas*, IEEE, 2016, p.1-12.

HEMMAPLARDH, A.; THURMAN, R. E. STAMATOYANNOPOULOS, J. A.; NOBLE, W. S. Unsupervised segmentation of continuous genomic data. *Bioinformatics Applications Note*, Washington, v. 23, n. 1, p. 1424-1426, fev, 2007.

HENNING, J.S.; DUSZA, S.W.; WANG, S.Q.; MARGHOOB, A.A.; RABINOVITZ, H.S. D.POLSKY, KOPF, A.W. The CASH (color architecture, symmetry, andhomogeneity)

algorithm for dermoscopy, *Journal of the American Academy of Dermatology*, v. 56, p. 45-52, jan. 2007.

HILAS, C.; Mastorocostas, P. An application of supervised and unsupervised learning approaches to telecommunications fraud detection. *Knowledge-Based System*, Amsterdam, v. 21, n. 7, p. 721-726, out, 2008.

HINTON, G. E.; SRIVASTAVA, N.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. R. Improving neural networks by preventing co-adaptation of feature detectors. Department Of Computer Scienci, University of Toronto, 2012, p.1-18. Disponível em: <<https://arxiv.org/pdf/1207.0580.pdf>>. Acesso em 9 de jul.2019.

HOFFMAN, M. M.; BUSKE, O. J.; WANG. J.; WENG. Z.; BILMES, J. A. NOBLE, W. S. Unsupervised pattern discovery in human chomatin structure through genomic segmentation. *Nature Mathods*. Washington, v.9, n. . p. 473-476. 2012.

HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the U.S.A.*, vol. 79, p.2554-2558, 1982.

HOPFIELD, J. J, TANK, D. W. Computing with neural circuits: A model. *Science*, vol. 233, p. 625-633, 1986.

JAIN, A. K.; JIANCHANG MAO; MOHIUDDIN, K. M. Artificial neural networks: a tutorial. *Computer*, v. 29, n. 3, p. 31–44, mar. 1996.

KASMI, R.; MOKRANI, K.; RADER, R. K.; COLE, J. G.; STOECKER, W. V. Biologically inspired skin lesion segmentation using a geodesic active contour technique. *Skin Research and Technology*, 2016.

KLAMBAUER, G.; UNTERHINER, T.; MAYR A.; HOCHREITER, S. Self-normalizing neural networks. *LIT AI Lab & Institute of Bioinfotmatics*, Linz, v. 5, p. 1-10. 2017.

KOHONEN, T. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*. Espoo. v. 43. p. 59-69. 1982.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet classification with deep convolutional neural networks. In: International Conference on Neural, 25, 2012, Lake Tahoe. *Anais...Lake Tahoe, Information Processing Systems*, dez. 2012.

LAW, M. A simple introduction to support vector machines. Lecture for CSE,802, 2006.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998.

LEVIN, S. Y.; WARSHAW, E. M. Teledermatology: A Review of Reliability and Accuracy of Diagnosis and Management. *Dermatologic Clinics*, v. 27, n. 2, p. 163–176, abr. 2009.

- LIBBRECHT, M. W.; NOBLE W. S. Machine learning applications in genetics and genomics. Epub 2015 Jun
- LIU, Z.; ZERUBIA, J. Skin image illumination modeling and chromophore identification for melanoma diagnosis. *Physics in Medicine and Biology*, 2015.
- LOANE, M.; GORE, H.; CORBET, R.; STEELE K. Effect of camera performance on diagnostic accuracy: Preliminary results from the Northern Ireland arms of the UK Multicentre Teledermatology Trial. *Journal of Telemedicine and Telecare*. vol. 3, n. 2, p. 83–88, 1997.
- MA, Z.; TAVARES, J. M. R. S. Effective features to classify skin lesions in dermoscopic images. *Expert Systems with Applications*, v. 84, p. 92–101, 30 out. 2017.
- MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. Computer Science Department, Stanford University, 2013 Disponível em:<<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.693.1422>>. Acesso em 9 jul. 2019.
- MAGLOGIANNIS, I.; DOUKAS, C. N. Overview of advanced computer vision systems for skin lesions characterization. *IEEE Transactions on Information Technology in Biomedicine*, v. 13, n. 5, p. 721–733, 2009.
- MAHBOD, A.; SCHAEFER, G.; WANG, C.; ECKER, R.; ELLINGER, I. Skin Lesion Classification Using Hybrid Deep Neural Networks. p. 1–5, 2017.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, v. 5, n. 4, p. 115–133, dez. 1943.
- MINSKY, M.; PAPERT, S. *Perceptrons: an introduction to computational geometry*. Oxford, The MIT Press, 1969.
- MEHTA, A.; PARIHAR, A. S.; MEHTA, N. Supervised classification of dermoscopic images using optimized fuzzy clustering based Multi-Layer Feed-forward Neural Network. In: International Conference on Computer, Communication and Control (IC4), Indore. *Anais...Indore: IEEE*, set. 2015, p. 1-6.
- MENZIES, S. W.; CROTTY K.A.; INGVAR, C.; MCCARTHY W. H. An atlas or surface microscopy of pigmented skin lesions. New York: McGraw-Hill, p. 1, 1996
- MIOTTO, R.; WANG, F.; WANG, S.; JIANG, X.; DUDLEY, J. T. Deep learning for healthcare: Review, opportunities and challenges. *Briefings in Bioinformatics*, v. 19, n. 6, p. 1236–1246, 2017.
- MISHRA, N. K.; CELEBI, M. E. An Overview of melanoma detection in dermoscopy images using image processing and machine learning. p. 1–15, 2016.
- NG, V. T. Y.; FUNG, B. Y. M.; LEE, T. K. Determining the asymmetry of skin lesion with

- fuzzy borders. *Computers in Biology and Medicine*, LOCAL, v. 35, p. 103-120, 2005.
- NWANKPA, C.; IJOMAH, W.; GACHAGAN, A.; MARSHALL, S. Activation functions: Comparison of trends in practice and research for deep learning. *Corr*, Glasgow, v.1811 p. 1–20, 2018.
- OGORZALEK, M.; SURÓWKA, G.; NOWAK, L.; MERKWIRTH, C. Computational Intelligence and image processing methods for applications in skin cancer diagnosis. In: *Biomedical Engineering Systems and Technologies*. BIOSTEC, v.52, 2009, Heidelberg, *Anais...* Heidelberg: Springer 2010. p. 3–20.
- OLIVEIRA, R. B.; MARRANGHELLO, N.; PEREIRA, A. S.; TAVARES, J. M. R. S. A computational approach for detecting pigmented skin lesions in macroscopic images. *Expert Systems with Applications*, Tarrytown, v. 61, p. 53-63, nov. 2016.
- OLIVEIRA, R. B.; PEREIRA, A. S.; TAVARES, J. M. R. S. Skin lesion computational diagnosis of dermoscopic images: Ensemble models based on input feature manipulation. *Computer Methods and Programs in Biomedicine*, v. 149, p. 43–53, out. 2017.
- PATHAN, S.; PRABHU, K. G.; SIDDALINGASWAMY, P. C. Techniques and algorithms for computer aided diagnosis of pigmented skin lesions. *Biomedical Signal Processing and Control*, v. 39, p. 237–262, 2018.
- PATWARDHAN, S. V.; DAI, S.; DHAWAN A. P. Multi-spectral image analysis and classification of melanoma using fuzzy membership based partitions. *Computerized Medical Imaging and Graphics*, Amsterdam: Elsevier, 2005.
- PENNISI, A.; BLOISI, D. D.; NARDI, D.; GIAMPETRUZZI, A. R.; MONDINO, C.; FACCHIANO, A. Skin lesion image segmentation using Delaunay Triangulation for melanoma detection. *Computerized Medical Imaging and Graphics*. Amsterdam: Elsevier, 2016.
- PICCOLI, M. F.; AMORIM, B. D. B.; WAGNER, H. M.; NUNES, D. H. Teledermatology protocol for screening of skin cancer. *Anais Brasileiros de Dermatologia*, Rio de Janeiro, v. 90, n. 2, p. 202–210, 2015.
- PITTARAS, N.; MARKATOPOULOU, F.; MEZARIS, V.; PATRAS, I. Comparison of Fine-tuning and extension strategies for deep convolutional neural networks. In: *International Conference on Multimedia Modeling*, 23, 2017. Londres. *Anais..* Reykjavik: MMM, jan. 2017. p.102-114.
- PLEISS, C.; RISSE, J. H.; BIRSACK, H.-J.; BENDER, H. Role of FDG-PET in the assessment of survival prognosis in melanoma. *Cancer Biother. Radiopharm*, v. 22, n. 6, p. 740–747, Dec. 2007.

- PREMALADHA, J.; RAVICHANDRAN, K. S. Novel approaches for diagnosing melanoma skin lesions through supervised and deep learning algorithms. *Journal of Medical Systems*, Thanjavur, v. 40, n. 4, Nov. p. 187-198, Apr.2016.
- RISH, I. An empirical study of the naive Bayes classifier. In: International Joint Conferences on Artificial Intelligence - Workshop on Empirical Methods in Artificial Intelligence, 3, 2001, New York. *Anais...* New York: Morgan Kaufmann Publishers Inc, US, 2001 p. 41-46.
- ROBBINS, H.; SIEGMUND, D. A convergence theorem for non negative almost supermartingales and some applications. In: ROBBINS, H.; SIEGMUND, D. *Optimizing Methods in Statistics*. Columbus: Elsevier, 1971. p. 233-257.
- RUMELHART, D. E.; HINTON, G. E; WILLIAMS; R. J. Learning representations by back-propagating errors. In: RUMELHART, D. E.; HINTON, G. E; WILLIAMS; R. J. *Neurocomputing: foundations of research*. Cambridge: MIT Press, 1988. p. 696-699.
- SCHEFFER, M.; CASSENOTE, A.; GUILLOUX, A. G. A.; MIOTTO, B. A.; MAINARDI, G. M. *Demografia Médica no Brasil. Brasilia 2018*. São Paulo, SP: FMUSP, Cremesp, p. 1-286. 2018.
- SCHERER, D.; MÜLLER, A.; BEHNKE, S. Evaluation of pooling operations in convolutional architectures for object recognition. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*, v. 6354 LNCS, n. PART 3, p. 92–101, 2010.
- SCHMITT, J. V.; MIOT, H. A.; SCHMITT, J. V.; MIOT, H. A. Distribution of Brazilian dermatologists according to geographic location, population and HDI of municipalities: an ecological study. *Anais Brasileiros de Dermatologia. Rio de Janeiro*. v. 89, n. 6, p. 1013–1015, dez. 2014.
- SERMANET, P., EIGEN, D., ZHANG, X., MATHIEU, M., FERGUS, R., LECUN, Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. In International Conference on Learning Representations, dez. 2013, New York. *Anais...* Banff: CBLS. apr.2014.
- SFORZA, G.; CASTELLANO, G.; ARIKA, S. K.; LEANDER, R. W.; STANLEY, R. J.; STOECKER, W. V.; HAGERTY, J. R. Using adaptive thresholding and skewness correction to detect gray areas in melanoma in situ images. *IEEE Transactions on Instrumentation and Measurement*. St. James, v. 61, n. 7. p. 1839-1847. may. 2012.
- SILVA, L. N. C.; ZUBEN, F. J. V. Análise e Síntese de Estratégias de Aprendizado para Redes Neurais Artificiais, Dissertação (mestrado), 1998.
- SHRIVASTAVA, V. K.; LONDHE, N. D.; SONAWANE, R. S.; SURI, J. S. Computer-aided

diagnosis of psoriasis skin images with HOS, texture and color features: A first comparative study of its kind. *Computer Methods and Programs in Biomedicine*, v. 126, p. 98–109, abr. 2016.

SIMONYAN, K.; ZISSERMAN, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. p. 1–14, 2014.

SONKA, M.; HLAVAC, V.; BOYLE, R. Image Processing, Analysis, and Machine Vision Second Edition. *Int'l Thomson*, 1999.

TAOUIL, K.; BEN ROMDHANE, N. Automatic segmentation and classification of skin lesion images. In: Proceedings of DFMA - International Conference on Distributed Frameworks for Multimedia Applications, 2, 2006. Palau. *Anais...* Palau: Institute of Electrical and Electronics Engineers - IEEE, jun. 2006. p. 1-12.

UMBAUGH, S. E.; MOSS, R. H. STOECKER, W. V. Applying artificial intelligence to the identification of variegated coloring in skin tumors. *Engineering in Medicine and Biology Magazine IEEE*, v. 10, n. 4, p. 57–62, Dec. 1991.

UMBAUGH, S. E.; MOSS, R. H.; STOECKER, W. V. An automatic color segmentation algorithm with application to identification of skin tumor borders: *Computerized Medical Imaging and Graphics*. Amsterdam: Elsevier, jun. 1992.

VAPNIK, V. *Statistical learning theory*. Canada: John Wiley & Sons, 1998.

VIEIRA, E. W. R.; LIMA, T. M. N.; GAZZINELLI, A. The waiting time for specialized medical consultations in a small municipality of Minas Gerais, Brazil. *REME: Revista Mineira de Enfermagem*, v. 19, n. 1, p. 65–71, 2015.

WALTER, F.M.; PREVOST, A.T.; VASCONCELOS, J.; HALL, P.N.; BURROWS, N.P.; MORRIS, H.C. Using the 7-point checklist as a diagnostic aid for pigmented skinlesions in general practice: a diagnostic validation study. *The British journal of general practice: The journal of the Royal College of General Practitioners*, v. 63, n. 610, p. 345-353. May. 2013.

WIDROW, B.; HOFF, M. E. Associative Storage and Retrieval of Digital Information in Networks of Adaptive Neurons, Biological Prototypes and Synthetic Systems, p. 1-160, 1962

WONG, A.; SCHARCANSKI, J.; FIEGUTH, P. Automatic skin lesion segmentation via iterative stochastic region merging. *Transactions on Information Technology in Biomedicine (IEEE)*, v. 15, n. 6, p. 929-936, 2011.

YOSINSKI, J.; CLUNE, J.; BENGIO, Y.; LIPSON, H. How transferable are features in deep neural networks? In: International Conference on Neural Information Processing Systems, 27, 2014, Montreal. *Anais...* Cambridge: MIT Press. Dez. 2014, p. 3320-3328.

ZEILER, M. D.; RANZATO, M.; MONGA, R.; MAO, M.; YANG, K.; LE, Q. V.;

NGUYEN, P.; SENIOR, A.; VANHOUCHE, V.; DEAN, J.; HINTON, G. E. On rectified linear units for speech processing. In: International Conference on Acoustics, Speech and Signal Processing - Proceedings, 38., 2013. Vancouver, *Anais...* Vancouver, IEEE, 2013 p. 3517-3521.

ZORTEA, M.; FLORES, E.; SCHARCANSKI, J. *Pattern Recognition: A simple weighted thresholding method for the segmentation of pigmented skin lesions in macroscopic images*. Porto Alegre: Elsevier, apr. 2017.

ZOURIDAKIS, G.; DOSHI, M.; MULLANI, N. Early diagnosis of skin cancer based on segmentation and measurement of vascularization and pigmentation in nervoscope images. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 26, sept. 2004, San Francisco. *Anais...* San Francisco: IEEE. 2005. p. 1593–1596.

9 ANEXOS

9.1 Anexo 1 Agrupamento de CIDs

CID	Agrupamento
L00, L01.0, L02.9, L03.9, I88.9, L08.0, L08.1, L30.3, L75.0, B43.9, B85.2, B35.0, B35.2, B35.3, B35.4, B35.5, B35.6, B35.9, B02, B00, B36.0, B35.1, L08.8, B37.9, N48.1, B86, L03.0	INFECÇÃO SUPERFICIAL
L05.0, L05.9, L72.0, L72.1, L72.2, L72.9, L72.8, D17.3	CISTO BENIGNO
L10.0, L10.1, L10.2, L10.3, L10.4, L10.5, L11.1, L12.0, L12.1, L12.2, L12.3, L13.0	LESÃO BOLHOSA
L11.0, L57.4, L64.9, L66.9, L67, L68.3, L85.0, L85.1, L85.2, L90.6, L60.2, L60.5, E70.3, R61.9	CAUSA GENÉTICA
L20.9, L21.9, L22, L23.9, L24.9, L28.0, L28.1, L29.9, L30.0, L30.1, L30.2, L30.4, L30.5, L56.2, L74.0, L74.1, L75.2, L85.3, I83.1, K13.0, L83.1	ECZEMA
L27.0, L27.1, L43.2, L51.2, L51, L56.0, L56.1, L53.8	REAÇÃO ADVERSA A DROGA
L40.4, L40.1, L40.2, L40.3, L40.5, L41.0, L41.1, L41.2, L41.3, L41.4, L41.5, L42, L43.0, L43.1, L43.3, L44.0, L44.1, L44.2, L44.3, L44.4, L43.9, L50.1, L50.2, L50.3, L50.4, L50.5, L50.6, L50.9, L52, P83.1, L53.2, L63.0, L63.1, L63.2, L63.9, L66.0, L66.1, L66.2, L66.3, L70.0, L70.1, L70.4, L70.9, L71.9, L25.1, L73.0, L73.2, L81.7, L88, L92.2, L95.0, L95.1, L95.9, L97, L98.2, L98.3, L98.4, E85.4, L53.1, L92.0	INFLAMAÇÃO DE CAUSA DESCONHECIDA
T30.1, T30.3, T30.2, L56.3, L56.4, L57.1, L57.2, L57.3, L57.5, L58.0, L58.1, L59.0, L73.1, L83, L84, L89, L90.5, L90.8, T81.5, L60.0, L60.1, L60.3	CAUSA TRAUMÁTICA
D04, L56.8	PRÉ-MALIGNO
L65.0, L65.1, L65.2, L65.9, L68.0, L87, L92.1, L60.4, E80.1,	CAUSA METABOLICA
L71.1, L82, L91.0, L98.0, B08.1, B07, D23.9, D18.0, D21.9, D22.9, Q82.8, A63.0, H02.6, D36.1	TUMOR BENIGNO
L81.0, L81.1, L81.2, L81.3, L81.5, 81.4, L80	DISTURBIO DE PIGMENTAÇÃO
L90.0, L93, L93.0, L93.1, L94.0, L94.1, L94.2, L94.4, L94.5, L94.6, M33.9	COLAGENOSE
L98.1	DISTÚRBIO PSIQUIÁTRICO
D04.9, C43.9, C80, C44.9, I78.1, Q82.5, C84.0	TUMOR MALIGNO
A30.9, B42.9, B55.9, B41.9	INFECÇÃO PROFUNDA