



UNIVERSIDADE
ESTADUAL DO
MARANHÃO



UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO E
SISTEMAS

LÚCIO FLÁVIO DE JESUS SILVA

**Combinando Redes Convolucionais no Auxílio do Diagnóstico da COVID-19 em
Tomografias Computadorizadas**

São Luís

2021

LÚCIO FLÁVIO DE JESUS SILVA

**Combinando Redes Convolucionais no Auxílio do Diagnóstico da COVID-19 em
Tomografias Computadorizadas**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Computação e Sistemas da Universidade Estadual do Maranhão, como parte dos requisitos para obtenção do título de Mestre em Engenharia de Computação e Sistemas.

Área de Concentração: Tecnologia da Informação.

Orientador: Omar Andrés Carmona Cortés

São Luís

2021

*Dedico esta dissertação aos meus heróis,
Gesilene e Lucas.
Sem eles, o caminho seria bem mais difícil.*

“É interessante notar que enquanto há relatos que alguns golfinhos aprenderam palavras – até cinquenta palavras utilizadas no contexto correto – não há relatos de seres humanos terem aprendido golfinês.”
Carl Sagan

AGRADECIMENTOS

Agradeço aos meus pais Gesilene e Lucas pelo carinho, atenção e apoio que eles me deram durante toda a minha vida. Sou grato aos meus amigos Luiza, Alysson e Dinailson por entenderem minha rotina e me apoiarem em todos os momentos. Agradeço também ao professor Dr. Omar Andrés Carmona Cortés por todo o apoio técnico prestado durante todo o desenvolvimento do projeto.

Resumo

SILVA, Lúcio Flávio de Jesus. **Combinando Redes Convolucionais no Auxílio do Diagnóstico da COVID-19 em Tomografias Computadorizadas**. 2021. Dissertação (Mestrado Profissional em Engenharia de Computação e Sistemas) – Centro de Ciências Tecnológicas, Universidade Estadual do Maranhão, São Luís, 2021.

A COVID-19 é uma doença infecciosa que causa uma síndrome respiratória aguda que pode levar a morte em poucos dias. Assim, a detecção rápida da doença pode proporcionar mais tempo para o tratamento, mesmo que ainda não haja um tratamento conhecido. Nesse contexto, este trabalho investiga quatro combinações de agregação de classificadores usando aprendizagem profunda e transferência de aprendizagem em imagens de tomografia computadorizada. A base de dados é formada por 2477 imagens divididas em duas classes para COVID-19: positiva e negativa. Foram testadas 11 redes neurais convolucionais (DenseNet121, DenseNet169, DenseNet201, VGG16, VGG19, Xception, ResNet50, ResNet50v2, Inception V3, MobileNet e MobileNetv2) para determinar as três melhores que foram combinadas em quatro diferentes combinações. Os resultados demonstraram que as três melhores redes neurais foram as DenseNet169, VGG16 e Xception. Além disso, verificou-se que a combinação das três redes, chamada de EnsembleDVX, levou aos melhores resultados tanto em todas as combinações quanto se comparada às CNNs testadas, chegando a uma precisão de 97,7%, recall igual a 97,7%, F1 Score de 97,8%, acurácia de 97,7% e uma perda de apenas 0.077.

Palavras-chave: Aprendizagem profunda. Aprendizagem por transferência. Aprendizagem por agrupamento.

Abstract

SILVA, Lúcio Flávio de Jesus. **Combining Convolutional Networks to Aid the Diagnosis of COVID-19 in Computed Tomography**. 2021. Dissertation (Professional Master in Computer and Systems Engineering) – Technological Sciences Center, Maranhão State University, São Luís, 2021.

COVID-19 is an infectious disease that causes an acute respiratory syndrome that can lead to death in a few days. Thus, the rapid detection of the disease can provide more time for treatment, even if there is no known treatment. In this context, this work investigates four combinations of classifier aggregation using deep learning and transfer of learning in computed tomography images. The database is formed by 2477 images divided into two classes for COVID-19: positive and negative. 11 convolutional neural networks (DenseNet121, DenseNet169, DenseNet201, VGG16, VGG19, Xception, ResNet50, ResNet50v2, Inception V3, MobileNet and MobileNetv2) were tested to determine the best three that were combined in four different combinations. The results showed that the three best neural networks were DenseNet169, VGG16 and Xception. In addition, it was found that the combination of the three networks, called EnsembleDVX, led to the best results both in all combinations and compared to the tested CNNs, reaching an precision of 97.7%, recall equal to 97.7%, F1 Score of 97.8%, accuracy of 97.7% and a loss of only 0.077.

Keywords: Deep Learning. Transfer Learning. Ensemble Learning.

Lista de figuras

Figura 1 – Neurônio biológico.....	21
Figura 2 – Regressão linear	22
Figura 3 – Representação MLP	24
Figura 4 – Representação CNN	26
Figura 5 – Gráfico EnsembleDVX	35
Figura 6 – Imagens Covid-19 padronizadas	37
Figura 7 – Imagens Non Covid-19 padronizadas	37
Figura 8 – Matriz de Confusão – DenseNet169	40
Figura 8.1 – Acurácia e Perda Fold 1 - DenseNet169	40
Figura 8.2 – Acurácia e Perda Fold 2 - DenseNet169	41
Figura 8.3 – Acurácia e Perda Fold 3 - DenseNet169	41
Figura 8.4 – Acurácia e Perda Fold 4 - DenseNet169	41
Figura 8.5 – Acurácia e Perda Fold 5 - DenseNet169	41
Figura 9 – Matriz de Confusão – VGG16	42
Figura 9.1 – Acurácia e Perda Fold 1 - VGG16	42
Figura 9.2 – Acurácia e Perda Fold 2 - VGG16	43
Figura 9.3 – Acurácia e Perda Fold 3 - VGG16	43
Figura 9.4 – Acurácia e Perda Fold 4 - VGG16	43
Figura 9.5 – Acurácia e Perda Fold 5 - VGG16	43
Figura 10 – Matriz de Confusão – Xception	44
Figura 10.1 – Acurácia e Perda Fold 1 - Xception	44
Figura 10.2 – Acurácia e Perda Fold 2 - Xception	45
Figura 10.3 – Acurácia e Perda Fold 3 - Xception	45
Figura 10.4 – Acurácia e Perda Fold 4 - Xception	45
Figura 10.5 – Acurácia e Perda Fold 5 - Xception	45
Figura 11 – Matriz de Confusão – DenseNet201	46
Figura 11.1 – Acurácia e Perda Fold 1 - DenseNet201	46
Figura 11.2 – Acurácia e Perda Fold 2 - DenseNet201	47
Figura 11.3 – Acurácia e Perda Fold 3 - DenseNet201	47
Figura 11.4 – Acurácia e Perda Fold 4 - DenseNet201	47
Figura 11.5 – Acurácia e Perda Fold 5 - DenseNet201	47
Figura 12 – Matriz de Confusão – DenseNet121	48

Figura 12.1 – Acurácia e Perda Fold 1 - DenseNet121	48
Figura 12.2 – Acurácia e Perda Fold 2 - DenseNet121	49
Figura 12.3 – Acurácia e Perda Fold 3 - DenseNet121	49
Figura 12.4 – Acurácia e Perda Fold 4 - DenseNet121	49
Figura 12.5 – Acurácia e Perda Fold 5 - DenseNet121	49
Figura 13 – Matriz de Confusão – VGG19	50
Figura 13.1 – Acurácia e Perda Fold 1 - VGG19	50
Figura 13.2 – Acurácia e Perda Fold 2 - VGG19	51
Figura 13.3 – Acurácia e Perda Fold 3 - VGG19	51
Figura 13.4 – Acurácia e Perda Fold 4 - VGG19	51
Figura 13.5 – Acurácia e Perda Fold 5 - VGG19	51
Figura 14 – Matriz de Confusão – ResNet50	52
Figura 14.1 – Acurácia e Perda Fold 1 - ResNet50	52
Figura 14.2 – Acurácia e Perda Fold 2 - ResNet50	53
Figura 14.3 – Acurácia e Perda Fold 3 - ResNet50	53
Figura 14.4 – Acurácia e Perda Fold 4 - ResNet50	53
Figura 14.5 – Acurácia e Perda Fold 5 - ResNet50	53
Figura 15 – Matriz de Confusão – ResNet50V2	54
Figura 15.1 – Acurácia e Perda Fold 1 - ResNet50V2	54
Figura 15.2 – Acurácia e Perda Fold 2 - ResNet50V2	55
Figura 15.3 – Acurácia e Perda Fold 3 - ResNet50V2	55
Figura 15.4 – Acurácia e Perda Fold 4 - ResNet50V2	55
Figura 15.5 – Acurácia e Perda Fold 5 - ResNet50V2	55
Figura 16 – Matriz de Confusão – InceptionV3	56
Figura 16.1 – Acurácia e Perda Fold 1 - InceptionV3	56
Figura 16.2 – Acurácia e Perda Fold 2 - InceptionV3	57
Figura 16.3 – Acurácia e Perda Fold 3 - InceptionV3	57
Figura 16.4 – Acurácia e Perda Fold 4 - InceptionV3	57
Figura 16.5 – Acurácia e Perda Fold 5 - InceptionV3	57
Figura 17 – Matriz de Confusão – MobileNet	58
Figura 17.1 – Acurácia e Perda Fold 1 - MobileNet	58
Figura 17.2 – Acurácia e Perda Fold 2 - MobileNet	59
Figura 17.3 – Acurácia e Perda Fold 3 - MobileNet	59
Figura 17.4 – Acurácia e Perda Fold 4 - MobileNet	59

Figura 17.5 – Acurácia e Perda Fold 5 - MobileNet	59
Figura 18 – Matriz de Confusão – MobileNetV2.....	60
Figura 18.1 – Acurácia e Perda Fold 1 - MobileNetV2.....	60
Figura 18.2 – Acurácia e Perda Fold 2 - MobileNetV2.....	61
Figura 18.3 – Acurácia e Perda Fold 3 - MobileNetV2.....	61
Figura 18.4 – Acurácia e Perda Fold 4 - MobileNetV2.....	61
Figura 18.5 – Acurácia e Perda Fold 5 - MobileNetV2.....	61
Figura 19 – Matriz de Confusão – EnsembleDVX	62
Figura 19.1 – Acurácia e Perda Fold 1 - EnsembleDVX	62
Figura 19.2 – Acurácia e Perda Fold 2 - EnsembleDVX	63
Figura 19.3 – Acurácia e Perda Fold 3 - EnsembleDVX	63
Figura 19.4 – Acurácia e Perda Fold 4 - EnsembleDVX	63
Figura 19.5 – Acurácia e Perda Fold 5 - EnsembleDVX	63

Lista de tabelas

Tabela 1 – Comparação do estado da arte.	18
Tabela 2 – Resultado das 12 CNNs.	39
Tabela 3 – Folds DenseNet169.....	40
Tabela 4 – Folds VGG16.....	42
Tabela 5 – Folds Xception.....	44
Tabela 6 – Folds DenseNet201.....	46
Tabela 7 – Folds DenseNet121.....	48
Tabela 8 – Folds VGG19.....	50
Tabela 9 – Folds ResNet50.....	52
Tabela 10 – Folds ResNet50V2.....	54
Tabela 11 – Folds InceptionV3.	56
Tabela 12 – Folds MobileNet.	58
Tabela 13 – Folds MobileNetV2.	60
Tabela 14 – Folds EnsembleDVX.	62
Tabela 14.1 – Agrupamento dois em dois EnsembleDVX.....	64

Lista de abreviaturas e siglas

AM	Aprendizagem de Máquina
BOLD	Dependente do Nível de Oxigenação Sanguínea
CAD	Diagnóstico Auxiliado por Computador
CNN	Redes Neurais Convolucionais
fMRI	Imagem por Ressonância Magnética Funcional
IA	Inteligência Artificial
MLP	Perceptron multicamadas
MRI	Imagem por Ressonância Magnética
OCR	Reconhecimento ótico de caracteres
RNA	Redes Neurais Artificiais
SVM	Máquinas de Vetores de Suporte
TC	Tomografia Computadorizada

Sumário

1	Introdução.....	11
1.1	<i>Definição do problema.....</i>	<i>12</i>
1.2	<i>Formulação de hipóteses</i>	<i>13</i>
1.3	<i>Objetivo</i>	<i>13</i>
1.3.1	Objetivo geral	13
1.3.2	Objetivos específicos	13
1.4	<i>Metodologia</i>	<i>14</i>
1.5	<i>Trabalhos relacionados.....</i>	<i>15</i>
2	Material e Métodos	19
2.1	<i>Aprendizagem de máquina.....</i>	<i>19</i>
2.1.1	Redes neurais artificiais	19
2.1.2	Inspiração da neurociência	20
2.1.3	Fundamentos.....	21
2.1.4	Paradigmas de aprendizagem	22
2.2	<i>Redes Multilayer Perceptrons.....</i>	<i>23</i>
2.3	<i>Aprendizagem profunda.....</i>	<i>24</i>
2.3.1	Redes neurais convolucionais.....	25
2.3.2	Arquitetura das redes neurais convolucionais	27
2.3.3	Ensemble learning	27
2.4	<i>Classificação de imagens.....</i>	<i>28</i>
2.4.1	Imagens tomográficas.....	29
2.4.2	Imagens funcionais	29
2.5	<i>Ferramentas e técnicas</i>	<i>30</i>
2.5.1	TensorFlow	30
2.5.2	Keras	30
2.5.3	NVIDIA cuDNN	31
2.5.4	Transfer learning.....	31
2.5.5	Callbacks	31
3	Definição dos modelos.....	33
3.1	<i>Definição do modelo EnsembleDVX.....</i>	<i>33</i>

4	Experimentos	36
4.1	<i>Ambiente.....</i>	36
4.2	<i>Conjunto de dados.....</i>	36
4.3	<i>Métricas de avaliação</i>	38
4.4	<i>Resultados</i>	39
5	Conclusão	65
5.1	<i>Trabalhos futuros.....</i>	66
	Referências	67

1 Introdução

Os Sistemas de Diagnóstico de Doenças Auxiliados por Computador (CAD) de acordo com Esteves, Lorena e Nascimento (2009) “geralmente envolvem um passo de classificação, que determina, por exemplo, a presença ou ausência de uma doença de interesse”. Assim, a detecção precoce de uma doença pode fornecer tempo suficiente para um tratamento ou ação bem sucedida.

Os sistemas CAD se enquadram em um campo denominado Medicina 4.0 (Wolf e Scholze, 2017), em que uma das aplicações inclui a combinação de tecnologias inovadoras de inteligência artificial, incluindo algoritmos de Aprendizagem de Máquina, com o objetivo de desenvolver sistemas de apoio à decisão clínica. Estes sistemas de apoio à decisão estão relacionados aos procedimentos para melhorar as decisões médicas, fornecendo com base em evidências, informações no momento do contato entre médico e paciente ou até mesmo na decisão do tratamento (Schnurr et al., 2018).

A Aprendizagem de Máquina (AM) segundo Monard e Baranauskas (2003) “é um campo de pesquisa da Inteligência Artificial que estuda o desenvolvimento de métodos capazes de extrair conceitos (conhecimento) a partir de amostras de dados”. Em geral, os algoritmos de AM são utilizados de forma a gerar classificadores para um conjunto de exemplos. Por classificação entende-se o processo de atribuir, a um determinado dado, o rótulo de uma (ou mais) classe(s) a qual ele pertence. Neste sentido, as técnicas de AM são empregadas na indução de um classificador que a partir de exemplos seja capaz de prever a classe de novos dados do montante em que foi treinado.

As abordagens de Aprendizagem de Máquina evoluíram para abordagens de Aprendizagem Profunda, que são uma forma mais poderosa e eficiente de lidar com as enormes quantidades de dados gerados a partir de abordagens modernas de descoberta (Xin et al., 2018). Entretanto, as redes neurais profundas, especialmente as Redes Neurais Convolucionais (CNNs) que trabalham com imagens, possuem muitas camadas e conexões, tornando-as computacionalmente complexas para o treinamento completo da rede. Porém, uma solução denominada aprendizagem por transferência pode ser aplicada para ajudar nesse problema. A aprendizagem por transferência também pode ser usada quando uma tarefa possui um número limitado de exemplos rotulados, a ideia é usar conhecimento prévio para resolver problemas semelhantes, como os humanos fazem. Assim, a aprendizagem por transferência é um método de reutilização de um modelo ou conhecimento para outra tarefa relacionada (Sarkar et al., 2018).

Além da aprendizagem por transferência, uma abordagem denominada de aprendizagem por agrupamento tem recebido atenção crescente na comunidade de inteligência computacional, pois sistemas que aplicam aprendizagem por agrupamento provaram ser eficazes e extremamente versáteis em um amplo espectro de domínios de problemas e aplicações do mundo real (Polikar, 2012). A aprendizagem por agrupamento se baseia na ideia de combinar diversos modelos de predição, treiná-los para uma mesma tarefa, e usá-los para produzir um modelo agrupado mais complexo, que é a soma de suas partes.

1.1 Definição do problema

O surto do coronavírus 2 2019 (COVID-19) da síndrome respiratória aguda grave (SARS-CoV-2) levou milhões de pessoas a serem infectadas e milhões de mortes em todo o mundo. A Organização Mundial da Saúde (OMS) divulgou o Relatório da Situação informando 110.974.862 casos e 2.460.792 mortes em todo o mundo até 22 de fevereiro de 2021 (World Health Organization, 2021). O surto fez com que sistemas de saúde entrassem em colapso e colocou em questão o preparado da sociedade diante de pandemias de vírus desconhecidos, onde o diagnóstico e o isolamento imediato poderiam evitar uma propagação em massa.

Os coronavírus (CoV) são uma família de vírus que causam doenças que vão do resfriado comum a enfermidades mais graves, como a Síndrome Respiratória do Oriente Médio (MERS-CoV) e a Síndrome Respiratória Aguda Grave (SARS-CoV). Pesquisas mostraram que o SARS-CoV foi transmitido de gatos Civet para humanos e MERS-CoV de camelos dromedários para humanos. Os coronavírus estão presentes em humanos e em outras espécies diferentes de animais (DE Wit et al., 2016).

Com o desenvolvimento de Sistemas de Diagnósticos de Doenças Auxiliados por Computador surgiu a possibilidade de determinar, por exemplo, a presença ou ausência de uma doença de interesse. Os algoritmos de aprendizagem profunda, principalmente as Redes Neurais Convolucionais (CNNs), vem sendo utilizados com sucesso no processamento e análise de imagens digitais, incluindo imagens médicas. O estudo sobre o desempenho de diversas arquiteturas de CNNs na análise e classificação de imagens médicas pode auxiliar profissionais e pesquisadores no desenvolvimento destes sistemas, aplicando a arquitetura que alcançou melhor desempenho na resolução de determinado problema.

Neste trabalho serão apresentadas onze arquiteturas de aprendizagem profunda conhecidas na literatura e propõe uma arquitetura denominada de EnsembleDVX que utiliza a abordagem de aprendizagem por agrupamento construída a partir de três arquiteturas diferentes

na tarefa de análise computacional de imagens médicas, no intuito de comparar os classificadores na tarefa de diagnóstico da COVID-19 analisando imagens de tomografia computadorizada. As arquiteturas utilizadas são: DenseNet121, DenseNet169, DenseNet201, Resnet50, Resnet50 v2, VGG16, VGG19, Mobilenet, Mobilenet v2, Xception e Inception v3. Para a realização dos experimentos foi utilizado um banco de dados composto por 2477 imagens de tomografia computadorizada, que é utilizada para detecção de doenças pulmonares, sendo 1250 de pacientes que testaram positivo para a COVID-19 e 1227 que tiveram diagnóstico negativo.

1.2 Formulação de hipóteses

Tendo em vista os avanços tecnológicos no campo da medicina, os algoritmos de aprendizagem profunda têm auxiliado profissionais da área de forma gradual na análise de imagens clínicas, porém, há uma carência de estudos que identifiquem quais as arquiteturas mais adequadas para análise dessas imagens. A combinação de redes convolucionais pode alcançar resultados superiores na análise de imagens quando comparada a uma única rede. Sendo assim, qual a melhor combinação de arquiteturas de aprendizagem profunda na análise de imagens médicas?

1.3 Objetivo

1.3.1 Objetivo geral

Analisar e comparar o desempenho da combinação entre diferentes arquiteturas de aprendizagem profunda aplicando a abordagem de aprendizagem por transferência no diagnóstico da COVID-19 em tomografias computadorizadas.

1.3.2 Objetivos específicos

Os objetivos específicos deste trabalho consistem em:

- Testar redes neurais convolucionais em imagens de tomografia computadorizada;
- Propor uma forma de combinação das três melhores redes neurais;
- Verificar o desempenho de todas as combinações das três redes.

1.4 Metodologia

Para a realização deste trabalho, o mesmo foi dividido em seis grandes atividades: Levantamento bibliográfico; Padronização dos dados; Treinamento das arquiteturas; Testes das redes neurais convolucionais em imagens de tomografia computadorizada; Combinação das três melhores redes neurais; Análise de desempenho das combinações propostas.

A busca desse conteúdo foi feita em livros de autores renomados e citados no âmbito de pesquisas relacionadas a aprendizagem de máquina e aprendizagem profunda, além de artigos científicos com publicações em revistas e sites universitários, foi utilizado também, buscadores como o Google Acadêmico. Além disso, foram utilizadas palavras específicas como forma de especificar a busca desse material de pesquisa, como: machine learning, deep learning, artificial neural networks, image classification, tomographic images, transfer learning e ensemble learning. Esses descritores geraram auxílio significativo no desenvolvimento da pesquisa científica.

A principal dificuldade encontrada foi na obtenção de um banco de dados robusto contendo imagens tomográficas de pacientes diagnosticados com uma patologia específica. Entretanto, existe um conjunto de dados denominado de SARS-COV-2 Ct-Scan Dataset (Soares et al., 2020), que uniu os pesquisadores aos dados do estudo e esse conjunto de dados foi utilizado para a o desenvolvimento desta pesquisa.

O SARS-COV-2 Ct-Scan Dataset está disponível ao público, contendo 1250 tomografias que são positivas para infecção por SARS-CoV-2 e 1227 tomografias de pacientes não infectados por SARS-CoV-2, 2477 tomografias computadorizadas no total. Esses dados foram coletados de pacientes reais em hospitais de São Paulo, Brasil. O objetivo deste conjunto de dados é incentivar a pesquisa e o desenvolvimento de métodos de inteligência artificial capazes de identificar se uma pessoa está infectada pelo SARS-CoV-2 por meio da análise de suas tomografias.

Uma equipe de pesquisadores da Qatar University, Doha, Qatar e da University of Dhaka, Bangladesh, juntamente com seus colaboradores do Paquistão e da Malásia em colaboração com médicos, também criaram um banco de dados de imagens de raios-X de tórax denominado de COVID-19 Radiography Database que contém casos positivos de COVID-19 juntamente com casos normais e imagens de Pneumonia Viral (M.E.H et al., 2020).

O COVID-19 Radiography Database também está disponível ao público, contém imagens de raios-x da Sociedade Italiana de Radiologia Médica e Intervencionista (SIRM) COVID-19 DATABASE (S.I et al., 2020), Novel Corona Virus 2019 Dataset desenvolvido por

(Cohen, Morrison e Dao, 2020). Além de imagens extraídas de 43 publicações diferentes contendo 1200 imagens que são positivas para infecção por SARS-CoV-2, 1341 de pacientes não infectados e 1345 de pacientes diagnosticados com Pneumonia Viral.

1.5 Trabalhos relacionados

O trabalho intitulado como "*Deep learning approaches for COVID-19 detection based on chest X-ray images*" aplicou abordagens baseadas em aprendizagem profunda, ou seja, extração de características profundas, ajuste fino de redes neurais convolucionais pré-treinadas (CNN) e treinamento ponta a ponta de um modelo CNN desenvolvido, foram usadas para classificar COVID-19 em imagens de raios-X. Para extração de recursos profundos, foram usados modelos CNN profundos pré-treinados (ResNet18, ResNet50, ResNet101, VGG16 e VGG19). Para a classificação dos recursos de profundidade, o classificador Support Vector Machines (SVM) foi usado com várias funções do kernel, Linear, Quadrático, Cúbico e Gaussiano. Os modelos CNN profundos pré-treinados acima mencionados também foram usados para o procedimento de ajuste fino. Um novo modelo CNN é proposto neste estudo com treinamento ponta a ponta. Um conjunto de dados contendo 180 imagens de pacientes com COVID-19 e 200 imagens de raios-X de tórax normais (saudáveis) foi usado na experimentação do estudo. A acurácia da classificação foi usada como medida de desempenho do estudo. Os trabalhos experimentais revelam que a aprendizagem profunda apresenta grande potencial na detecção da COVID-19 com base em imagens de raios-X de tórax. As características profundas extraídas do modelo ResNet50 e do classificador SVM com a função Linear kernel produziram uma pontuação de acurácia de 94,7%, que foi a maior entre todos os resultados obtidos. A obtenção do modelo ResNet50 ajustado foi de 92,6%, enquanto o treinamento de ponta a ponta do modelo CNN desenvolvido produziu um resultado de 91,6% (Ismael e Şengür, 2021).

Já no trabalho intitulado como "*Artificial Intelligence Distinguishes COVID-19 from Community Acquired Pneumonia on Chest CT*" foi realizado um estudo retrospectivo e multicêntrico, um modelo de aprendizado profundo, rede neural de detecção COVID-19 (COVNet), foi desenvolvido para extrair características visuais de exames tomográficos volumétricos de tórax para a detecção de COVID-19. Pneumonia adquirida na comunidade (PAC) e outros exames de tomografia computadorizada diagnosticados sem pneumonia foram incluídos para testar a robustez do modelo. O desempenho do diagnóstico foi avaliado pela área sob a curva de característica de operação do receptor (AUC), sensibilidade e especificidade (Li et al, 2020). O conjunto de dados coletado consistiu em 4356 exames de TC de tórax de 3322

pacientes. A idade média é de 49 ± 15 anos e havia um pouco mais pacientes do sexo masculino do que do sexo feminino (1838 vs 1484; valor de $p = 0,29$). A sensibilidade e especificidade por exame para detectar COVID-19 no conjunto de teste independente foi 114 de 127 (90% [IC 95%: 83%, 94%]) e 294 de 307 (96% [IC 95%: 93%, 98%]), respectivamente, com uma AUC de 0,96 (p -valor $<0,001$). A sensibilidade e especificidade por exame para detectar CAP no conjunto de teste independente foi de 87% (152 de 175) e 92% (239 de 259), respectivamente, com uma AUC de 0,95 (IC de 95%: 0,93, 0,97) (Li et al, 2020).

O trabalho intitulado “*SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification*” disponibilizou um banco de dados ao público, contendo 1250 tomografias que são positivas para infecção por SARS-CoV-2 e 1227 tomografias de pacientes não infectados por SARS-CoV-2, 2477 tomografias computadorizadas no total. Esses dados foram coletados de pacientes reais em hospitais de São Paulo, Brasil. O objetivo desta pesquisa foi incentivar a pesquisa e o desenvolvimento de métodos de inteligência artificial capazes de identificar se uma pessoa está infectada pelo SARS-CoV-2 por meio da análise de suas tomografias. Como resultado de linha de base para este conjunto de dados, foi usada a abordagem de Aprendizado Profundo eXplainable (xDNN) que alcançou F1 score de 97,31%, o que é muito promissor (Angelov e Soares, 2020).

Uma equipe de pesquisadores da Qatar University, Doha, Qatar e da University of Dhaka, Bangladesh, juntamente com seus colaboradores do Paquistão e da Malásia em colaboração com médicos, criaram um banco de dados de imagens de raios-X de tórax denominado de COVID-19 Radiography Database que contém casos positivos de COVID-19 juntamente com casos normais e imagens de Pneumonia Viral (M.E.H et al., 2020). A equipe alcançou resultados que foram publicados no artigo intitulado de “*Can AI help in screening Viral and COVID-19 pneumonia?*” onde o objetivo do artigo foi detectar automaticamente pacientes com pneumonia COVID-19 usando imagens de raios-X digitais, maximizando a precisão na detecção usando pré-processamento de imagem e técnicas de aprendizado profundo. Um conjunto de treinamento de imagem aumentada foi criado com 2500 imagens de cada categoria para treinamento e validação de quatro diferentes redes neurais convolucionais (CNNs) profundas pré-treinadas. Estas redes foram testadas para a classificação de dois esquemas diferentes (normal e pneumonia por COVID-19; normal, viral e pneumonia por COVID-19). A acurácia, sensibilidade, especificidade e precisão da classificação para ambos os esquemas foram 98,3%, 96,7%, 100%, 100% e 98,3%, 96,7%, 99%, 100%, respectivamente (M.E.H et al., 2020).

Já o trabalho intitulado como “*Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks*” apresentou a necessidade de implementar um sistema de detecção automática como uma alternativa de diagnóstico rápido para evitar a propagação do COVID-19 entre as pessoas. Neste estudo, cinco modelos baseados em rede neural convolucional pré-treinados (ResNet50, ResNet101, ResNet152, InceptionV3 e Inception-ResNetV2) foram propostos para a detecção de pacientes infectados com pneumonia por coronavírus usando radiografias de tórax. Implementamos três classificações binárias diferentes com quatro classes (COVID-19, normal (saudável), pneumonia viral e pneumonia bacteriana) usando a validação cruzada de 5 vezes. Considerando os resultados de desempenho obtidos, verificou-se que o modelo ResNet50 pré-treinado oferece o melhor desempenho de classificação (96,1% de acurácia para o Dataset-1, 99,5% de acurácia para o Dataset-2 e 99,7% de acurácia para o Dataset-3) entre outros quatro usados modelos (Narin, Kaya e Pamuk, 2020).

No trabalho intitulado como “*Deep Learning-based Detection for COVID-19 from Chest CT using Weak Label*” foi desenvolvido um sistema de software baseado em aprendizado profundo fracamente supervisionado usando volumes de TC 3D para detectar COVID-19. Para cada paciente, a região pulmonar foi segmentada usando uma UNet pré-treinada; em seguida, a região pulmonar 3D segmentada foi alimentada em uma rede neural profunda 3D para prever a probabilidade de COVID-19 infecciosa. 499 volumes de CT coletados de 13 de dezembro de 2019 a 23 de janeiro de 2020 foram usados para treinamento e 131 volumes de CT coletados de 24 de janeiro de 2020 a 6 de fevereiro de 2020, foram usados para teste. O algoritmo de aprendizado profundo obteve 0,959 ROC AUC e 0,976 PR AUC. Houve um ponto operacional com sensibilidade de 0,907 e especificidade de 0,911 na curva ROC. Ao usar um limite de probabilidade de 0,5 para classificar COVID-positivo e COVID-negativo, o algoritmo obteve uma acurácia de 0,901, um valor preditivo positivo de 0,840 e um valor preditivo negativo muito alto de 0,982. O algoritmo levou apenas 1,93 segundos para processar o volume de CT de um único paciente usando uma GPU dedicada. O modelo de aprendizado profundo supervisionado de forma fraca pode prever com precisão a probabilidade de infecção do COVID-19 em volumes de TC de tórax sem a necessidade de anotar as lesões para treinamento. O algoritmo de aprendizado profundo de alto desempenho e fácil treinamento fornece uma maneira rápida de identificar pacientes COVID-19, o que é benéfico para controlar o surto de SARS-CoV-2 (Zheng et al, 2020).

Por último, no trabalho intitulado como “*Deep learning COVID-19 detection bias: accuracy through artificial intelligence*” foi desenvolvido um modelo de aprendizado profundo para melhorar a precisão dos casos relatados e para prever com precisão da doença a partir de

radiografias de tórax. O modelo baseou-se em redes neurais convolucionais (CNNs) para detectar anormalidades estruturais e categorização de doenças que foram fundamentais para descobrir padrões ocultos. Para fazer isso, uma abordagem de aprendizagem de transferência foi implantada para realizar detecções nas radiografias ântero-posteriores do tórax de pacientes. Os resultados alcançaram acurácia de (96,3%) e perda (0,151 entropia cruzada binária) usando o conjunto de dados público composto por pacientes de diferentes países em todo o mundo. Como a matriz de confusão indica, o modelo foi capaz de identificar com precisão verdadeiros negativos (74) e verdadeiros positivos (32); este modelo de aprendizado profundo identificou três casos de falso positivo e um achado de falso negativo nas varreduras de pacientes saudáveis. O modelo desenvolvido minimiza a interação manual dependente de radiologistas, pois automatiza a identificação de anormalidades estruturais nos CXRs do paciente, e o modelo de aprendizado profundo pode detectar verdadeiros positivos e verdadeiros negativos e eliminar falsos positivos e falsos negativos com boa precisão (Vaid, Kalantar e Bhandari, 2020).

A Tabela 1 exibe a comparação entre os resultados dos diferentes trabalhos citados anteriormente.

Tabela 1 - Comparação do estado da arte.

Método	Images	COVID-19	Outras	Saudáveis	Total	Acurácia
Ismael & Şengür, 2021.	X-ray	180	n.a.	200	318	0.947
Li et al, 2020.	CT	1296	1735	1325	4356	0.960
Angelov & Soares, 2020.	CT	1252	n.a.	1230	2482	0.973
M.E.H et al., 2020.	X-ray	2500	n.a.	2500	5000	0.983
Narin et al, 2020.	X-ray	341	n.a.	2800	3141	0.961
Zheng et al, 2020.	CT	313	n.a.	229	542	0.901
Vaid et al, 2020.	X-ray	181	n.a.	364	545	0.963
Nosso método	CT	1250	n.a.	1227	2477	0.977

Fonte: Lúcio Silva, 2021.

2 Material e Métodos

2.1 *Aprendizagem de máquina*

Pesquisas de Aprendizagem de Máquina buscam o desenvolvimento de técnicas capazes de extrair conhecimento a partir de exemplos de dados (Mitchell, 1997). Encontram-se algoritmos de Aprendizagem de Máquina do qual o intuito é possibilitar que, depois de um determinado treinamento com um certo aglomerado de dados pelos quais as instâncias têm classificação familiar, uma máquina seja apta a interpretar novas informações e classificá-las de forma adequada a partir de uma generalização do que lhe foi mostrado anteriormente. Segundo Libralão et al. (2003) existem três paradigmas que sugerem a forma de aprendizado do algoritmo de Aprendizagem de Máquina:

- Supervisionado: necessita de supervisores para alcançar o melhor modelo pretendido na fase de treinamento;
- Não-supervisionado: possibilita que o algoritmo de AM aprenda a agrupar as entradas levando em consideração uma medida de similaridade entre elas;
- Por reforço: o aprendizado acontece por meio de gratificações, de acordo com o desempenho alcançado.

Ainda de acordo com Libralão et al. (2003) boa parte dos algoritmos de AM possuem como inspiração os sistemas biológicos (Redes Neurais Artificiais (RNAs) e os Algoritmos Genéticos), o aprendizado simbólico (Árvores de Decisão), os processos cognitivos (Raciocínio Baseado em Casos), e as Teorias Estatísticas (SVMs).

2.1.1 Redes neurais artificiais

RNAs é uma área de pesquisa que trabalha com a simulação de aptidões cognitivas do ser humano. Desenvolvem-se máquinas capazes de apresentar uma conduta inteligente, como se fossem comportamentos humanos. A inteligência humana é a mais desenvolvida dentre outras criaturas e a região responsável por esta inteligência nos humanos é o cérebro (Rauber, 2005).

Ainda de acordo com Rauber (2005), os elementos básicos são os neurônios, interconectados em redes o que concede o compartilhamento de informação entre eles, formando a inteligência biológica. Um interesse óbvio que surge desses fatos é o esforço de copiar a forma como funciona o cérebro em um ambiente computacional. Isso significa que a

pesquisa tenta compreender o funcionamento da inteligência habitada nos neurônios e esquematiza-la em uma estrutura artificial, por exemplo, uma junção de hardware e software, assim convertendo as redes neurais biológicas em redes neurais artificiais.

2.1.2 Inspiração da neurociência

Existem particularidades do cérebro humano que o habilitam ter um comportamento inteligente. Os tópicos seguintes refletem as mais significativas características que são principalmente atrativas para serem utilizadas em uma rede neural artificial, segundo Rauber (2005) são:

- Robustez e tolerância a falhas: A exclusão de neurônios não abala a funcionalidade geral;
- Capacidade de aprendizagem: O cérebro possui a capacidade de aprender novas tarefas que jamais foram executadas anteriormente;
- Processamento de informação incerta: Mesmo que o conhecimento fornecido esteja incompleto, acometido por ruído ou relativamente contraditório, ainda um raciocínio preciso é possível;
- Paralelismo: Um enorme número de neurônios está ativo simultaneamente. Não há restrição de um processador que impreterivelmente trabalhe uma informação após a outra.

Na Figura 1 mostra-se o modelo resumido de um neurônio real. O neurônio é uma célula que possui núcleo e corpo onde comportamentos químicos e elétricos demonstram o tratamento de informação. A saída da informação da soma é efetuada por impulsos elétricos que se difundem pelo axônio. No final do axônio encontram-se inúmeras ramificações que compartilham a informação para os neurônios vizinhos. A junção com outros neurônios é feita através de sinapses que estão interligadas a uma dendrite do neurônio receptor. A sinapse envia uma substância química quando for alertada pelo impulso do axônio. A substância se propaga entre sinapse e dendrite realizando o acoplamento entre dois neurônios vizinhos. Conforme os alertas (ou inibições) que células vizinhas enviam para a célula em consideração, ela trata a informação mais uma vez e a transmite através do seu axônio (Rauber, 2005).

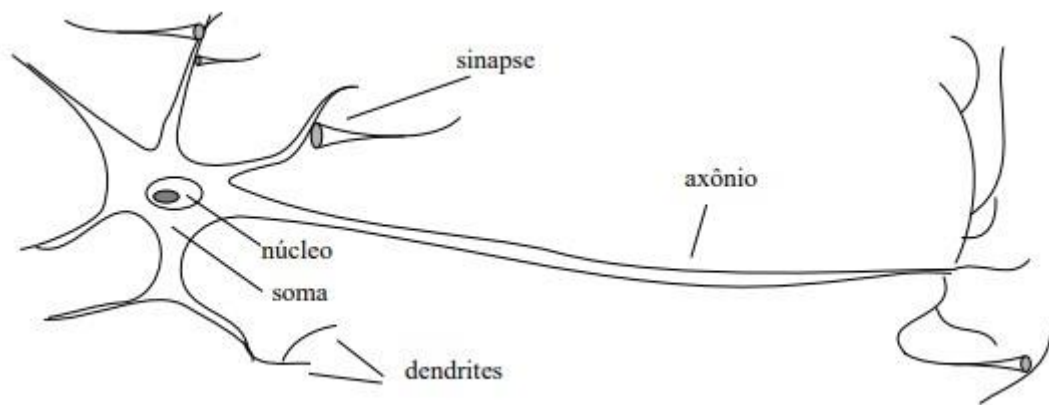


Figura 1 – Neurônio biológico (Raubert, 2005).

Um ponto importante na história das RNAs foi a demonstração de um modelo de um neurônio artificial por (McCulloch e Pitts, 1943). Os trabalhos nessa linha de pesquisa findaram na concepção do Perceptron por (Rosenblatt, 1958) e em um modelo similar, o Adaline por (Widrow and Hoff, 1960). O Perceptron possui a capacidade de classificar entre classes que são linearmente separáveis. Foi utilizado para reconhecer caracteres, por exemplo. Essa aplicação foi executada em uma máquina chamada MARK I PERCEPTRON e provocou uma grande euforia certamente exorbitante em relação a imaginação das habilidades de futuros robôs inteligentes.

2.1.3 Fundamentos

As Redes Neurais Artificiais (RNAs) aplicam um modelo matemático influenciado na arquitetura neural dos seres vivos, contraindo conhecimento através da experiência (Prampetro, 1998).

Tatibana e Kaetsu (2000) declaram que “as Redes Neurais Artificiais são formadas por um agrupado de nodos, que imitam o papel dos neurônios, ligados por um princípio de propagação. Cada nodo adquire suas entradas com os pesos relacionados, provenientes de outros nodos, ou de uma interação externa. A camada de entrada tem um nodo específico, chamado de bias, que ajuda aumentar os graus de liberdade, possibilitando uma melhor adaptação da rede”.

Ainda sobre estas entradas, é empregada uma função de ativação, que usa como argumento, nas entradas da rede, uma somatória ponderada. “O ponto de ativação de um nodo é definido pela função de ativação, normalmente uma função sigmoidal ou uma função degrau” (Perez et al., 1994). Uma camada que adquire sinais do ambiente exterior é chamada de camada

de entrada e uma camada que exprime sinais para o ambiente é dita camada de saída. Todas as outras são definidas camadas ocultas e não se ligam diretamente com o ambiente (Carvalho, 1999).

2.1.4 Paradigmas de aprendizagem

De acordo com Rauber (2005) uma vez determinada a rede neural, essa deve ser treinada. Isso significa que os graus de autonomia que a rede dispõe, para resolver a tarefa em consideração, devem ser adaptados da melhor maneira. Normalmente, isso significa que se deve alterar os pesos w_{ij} entre o neurônio i e o neurônio j , segundo um algoritmo. Um conjunto finito T de n amostras de treino está ao alcance para adaptar os pesos no decorrer do treinamento da rede. Uma distinção fundamental em relação aos paradigmas de aprendizagem que é pertinente para todo tipo de sistemas com técnica de adaptação é aprendizagem supervisionada, aprendizagem não-supervisionada e aprendizagem por reforço.

Na aprendizagem supervisionada cada amostra de treino está acompanhada por um custo que é o custo desejado. Isso significa que o conjunto de treino T está composto por n pares de amostras $(x_p y_p)$ onde $T = \{(x_p y_p)\}_{n_p = 1}$. A dimensão do vetor de entrada é D , *i.e.* as variáveis de entrada estão associadas em um valor multidimensional, normalmente faz parte dos números reais: $x = (x_1 \dots x_j \dots x_D)^T$, $x_j \in \mathbb{R}$. (A transposição $(.)^T$ é utilizada para economizar espaço registrando um vetor em uma linha). As variáveis de saída estão reunidas em um vetor de saída $y = (y_1 \dots y_i \dots y_c)^T$ (Rauber, 2005).

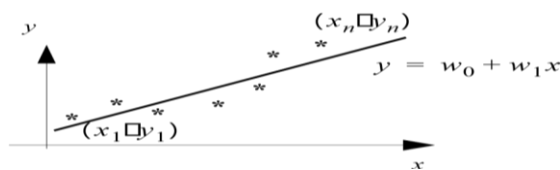


Figura 2 – Regressão linear (Rauber, 2005).

Segundo Rauber (2005) uma demonstração de uma tarefa de aprendizagem supervisionada é a regressão linear. Utiliza-se o caso unidimensional para simplificar a ilustração, como mostrado na Figura 2. Nesse problema o conjunto de treino constitui-se em pares de números reais $(x_p y_p)$. O objetivo da aprendizagem é a delimitação de coeficientes w_0 e w_1 da reta $y = w_0 + w_1 x$. O algoritmo de aprendizagem fomenta minimizar a discrepância entre o valor desejado y_p e o valor que é a resposta $y' = w_0 + w_1 x_p$ do sistema, e isso em média para cada amostra $(x_p y_p)$.

Já na aprendizagem não-supervisionada quando há uma única informação disponível, e essa informação são os valores (x_p) a tarefa de aprendizagem é encontrar correlações entre os exemplos de treino $T = \{(x_p)\}_{n_p = 1}$. O número de grupos ou classes não está definido a priori. Isso quer dizer que a rede tem que achar atributos estatísticos consideráveis, ela tem que desenvolver uma interpretação própria dos estímulos que entram na rede. Um sinônimo para aprendizagem não-supervisionada é aglomeração (“*clustering*”) (Rauber, 2005).

Um exemplo da área médica é a detecção de doenças a partir de imagens, como exemplo imagens de raio-X. Existem locais dentro da imagem que se deixam atribuir ao mesmo material, como exemplo, o osso. O número dos materiais (das aglomerações) é desconhecido a priori. O objetivo do sistema é achar o número dos materiais diferentes e ao mesmo tempo agrupar cada ponto da imagem para o respectivo material. A entrada para a rede poderiam ser os pontos da imagem, por exemplo um pequeno intervalo de 5 por 5 pontos. A resposta esperada da rede seria o material a qual pertence essa região da imagem (Rauber, 2005).

Na aprendizagem por reforço é realizado o treinamento de modelos de aprendizagem de máquina para tomar uma sequência de decisões. O agente aprende a alcançar um objetivo em um ambiente desconhecido e possivelmente complexo. Na aprendizagem por reforço, o sistema de aprendizagem de máquina enfrenta uma situação. Utilizando a abordagem de tentativa e erro para encontrar uma solução para um problema. Para cumprir a meta, são distribuídas recompensas ou penalidades pelas ações executadas. Seu objetivo é maximizar a recompensa total.

Embora o pesquisador defina a política de recompensa, ou seja, as regras do jogo, ele não oferece dicas ou sugestões de como resolver o jogo. Cabe ao modelo descobrir como executar a tarefa para maximizar a recompensa, começando com testes totalmente aleatórios e terminando com táticas avançadas. Ao alavancar o poder da pesquisa e de muitas tentativas, o aprendizado por reforço é atualmente a maneira mais eficaz de sugerir a criatividade da máquina (Data Science Academy, 2019).

2.2 *Redes Multilayer Perceptrons*

Um Perceptron é um classificador linear, ou seja, é um algoritmo que classifica a entrada separando duas categorias com uma linha reta. A entrada geralmente é um vetor de recursos x multiplicado por pesos w e adicionado a um viés (ou bias) b . Um Perceptron produz uma única saída com base em várias entradas de valor real, formando uma combinação linear usando os pesos (Data Science Academy, 2019).

Rosenblatt construiu um Perceptron de uma camada, ou seja, seu algoritmo não inclui múltiplas camadas, o que permite que as redes neurais modelem uma hierarquia de recursos. Isso impede que o Perceptron consiga realizar classificação não linear, como a função XOR (Minsky e Papert, 2017).

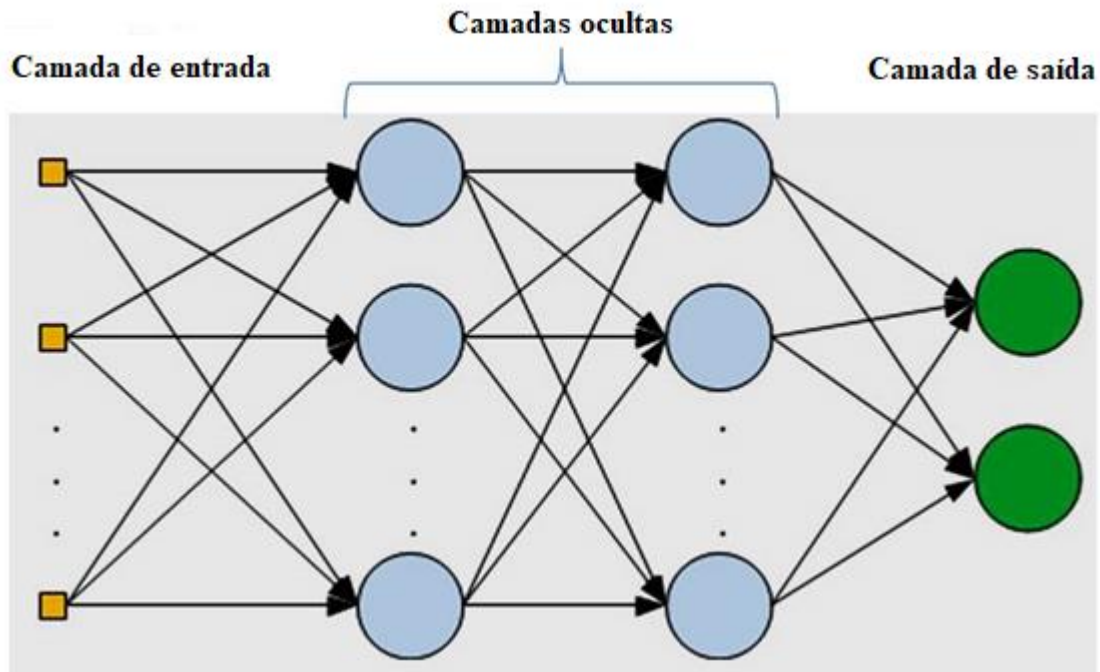


Figura 3 – Representação MLP (Data Science Academy, 2019).

Um Multilayer Perceptron (MLP) é uma rede neural artificial composta por mais de um Perceptron. Como pode ser observado na Figura 3, eles são compostos por uma camada de entrada para receber o sinal, uma camada de saída que toma uma decisão ou previsão sobre a entrada, e entre esses dois, um número arbitrário de camadas ocultas que são o verdadeiro mecanismo computacional do MLP. MLPs com uma camada oculta são capazes de aproximar qualquer função contínua.

2.3 *Aprendizagem profunda*

A aprendizagem profunda permite que modelos computacionais compostos por várias camadas de processamento aprendam representações de dados com vários níveis de abstração. Segundo Lecun (2015) esses métodos melhoraram drasticamente o estado da arte em reconhecimento de fala, reconhecimento de objetos, detecção de objeto e muitos outros domínios. O aprendizado profundo descobre a estrutura intrincada em grandes conjuntos de dados usando o algoritmo de Backpropagation para indicar como uma máquina deve alterar

seus parâmetros internos, que são usados para calcular a representação em cada camada a partir da representação na camada anterior. Redes convolucionais profundas trouxeram avanços no processamento de imagens, vídeo, voz e áudio, enquanto as redes recorrentes iluminaram dados sequenciais, como texto e fala.

2.3.1 Redes neurais convolucionais

As Redes Neurais Convolucionais (CNNs) são uma variação dos Perceptrons Multicamadas (MLP). Por uma inspiração biológica, as CNNs emulam o mecanismo básico do córtex visual animal. Os neurônios nas CNNs compartilham pesos ao contrário dos MLPs, onde cada neurônio possui um vetor de peso separado (Melo, 2018).

As Redes Neurais Convolucionais são redes neurais artificiais profundas que podem ser usadas para classificar imagens, agrupá-las por similaridade e realizar o reconhecimento. São algoritmos que podem identificar rostos, objetos e muitos outros aspectos visuais.

As CNNs realizam o reconhecimento óptico de caracteres (OCR) para digitalizar texto e tornar viável o processamento de linguagem natural em documentos analógicos e manuscritos, onde as imagens são símbolos a serem reconhecidos. As CNNs também podem ser aplicadas na análise de arquivos de áudio quando são representados visualmente como espectrograma.

As CNNs recebem imagens como volumes, ou seja, objetos tridimensionais, em vez de estruturas planas a serem medidas por largura e altura. Isso ocorre devido as imagens digitais possuírem uma codificação vermelho-verde-azul (RGB), combinando essas três cores para produzir o espectro de cores que os seres humanos percebem. Uma rede neural convolucional recebe imagens como três estratos separados de cores empilhados um em cima do outro (Data Science Academy, 2019).

Sendo assim, uma CNN recebe uma imagem como uma caixa retangular das quais a largura e altura são medidas pelo número de pixels ao longo dessas dimensões e cuja profundidade é de três camadas profundas, uma para cada letra em RGB. Essas camadas de profundidade são ditas como canais.

Segundo a Data Science Academy (2019) à medida que as imagens se movem através de uma CNN, são denominadas de volumes de entrada e saída, expressando-as matematicamente como matrizes de múltiplas dimensões. De camada em camada, suas

dimensões mudam à medida que atravessam a CNN, como pode ser observado na Figura 4, até gerar uma série de probabilidades na camada de saída, sendo uma probabilidade para cada possível classe de saída. Aquela com maior probabilidade, será a classe definida para a imagem de entrada.

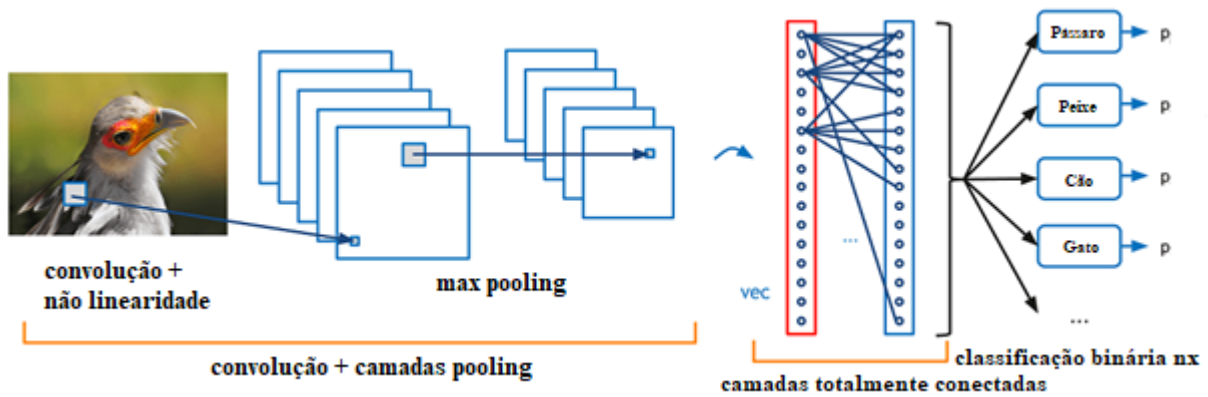


Figura 4 – Representação CNN (Data Science Academy, 2019).

Utilizando a estratégia de compartilhamento de pesos, os neurônios são capazes de realizar convoluções nos dados com o filtro de convolução formado pelos pesos. Isto é seguido por uma operação de *pooling* que é uma forma de subamostragem não-linear que reduz progressivamente o tamanho espacial da representação, reduzindo assim a quantidade de parâmetros e computação na rede (Melo, 2018).

Após várias camadas de convolução e *pooling*, o tamanho da matriz de entrada (tamanho do mapa de características) é reduzido e características mais complexas são extraídas. Eventualmente, com um mapa de características pequeno o suficiente, o conteúdo é comprimido em um vetor de uma dimensão e é alimentado em um MLP totalmente conectado para processamento (Melo, 2018).

Normalmente, entre a camada de convolução e a camada de *pooling* é aplicada uma camada ReLu na qual uma função de ativação não saturada de um lado é aplicada elemento a elemento, como por exemplo, $f(x) = \max(0, x)$, limitando os menores valores de saída da camada a zero (Melo, 2018).

A última camada do MLP totalmente conectado, visto como a saída, é uma camada de perda que é usada para especificar como o treinamento na rede penaliza o desvio entre os rótulos previstos e verdadeiros (Melo, 2018).

A eficácia das redes convolucionais no reconhecimento de imagem é uma das principais razões pelas quais o mundo testemunhou a eficácia do aprendizado profundo. Este tipo de rede

está impulsionando grandes avanços em Visão Computacional, que tem aplicações em carros autônomos, robótica, drones, segurança e diagnósticos médicos.

2.3.2 Arquitetura das redes neurais convolucionais

Uma arquitetura CNN é construída por três componentes: camada convolucional, camada de pooling e uma totalmente conectada. As camadas convolucionais usam filtros que examinam uma parte da imagem e extraem recursos dela. Esses recursos são geralmente cores, formas e bordas que definem uma imagem específica (Beysolow II, 2017). CNNs podem ter quantas camadas convolucionais forem necessárias. Quanto mais camadas convolucionais, mais recursos são extraídos.

A convolução unidimensional é uma operação entre um vetor de pesos $m \in R^m$ e um vetor de entradas vistas como uma sequência $s \in R^S$. O vetor m é o filtro da convolução. Concretamente, pensamos em s como a sentença de entrada e $s_i \in R$ é um valor de característica única associado à i -ésima palavra na sentença. A ideia por trás da convolução unidimensional é pegar o produto escalar do vetor m com cada m -grama na sentença para obter outra sequência c conforme é mostrado na Equação 1.

$$c_j = m^t s_{j-m+1:j} \quad (1)$$

Após as camadas convolucionais, adicionamos camadas de pooling que são responsáveis por agrupar os mapas de recursos em uma imagem (Beysolow II, 2017), proporcionando uma redução de dimensionalidade, que é produzida pela aplicação de um único máximo ou média dos valores dentro de uma caixa produzida pela camada convolucional. Assim, as camadas totalmente conectadas, que são redes MLP normais, recebem uma imagem menor do que aquela apresentada na entrada da CNN.

2.3.3 Ensemble learning

Ensemble learning ou aprendizagem por agrupamento refere-se aos procedimentos empregados para treinar vários modelos de aprendizagem e combinar seus resultados, tratando-os como um “comitê” de tomadores de decisão. O princípio é que previsões individuais combinadas de forma adequada, podem alcançar melhores precisões, em média, do que qualquer modelo individual.

Segundo Brown (2010) numerosos estudos empíricos e teóricos têm demonstrado que os modelos de agrupamento muitas vezes atingem maior precisão do que os modelos individuais.

Os membros do agrupamento podem estar prevendo números com valor real, rótulos de classe, posterior probabilidades, classificações ou qualquer outro dado. Portanto, suas decisões podem ser combinadas por muitos métodos, incluindo cálculo de média, votação e métodos probabilísticos.

Esse agrupamento objetiva, ao juntar múltiplos modelos mais fracos, diminuir a suscetibilidade geral deles a erros, tornando-os mais robustos. Portanto, os métodos de agrupamento devem levar em conta a maneira com a qual eles agrupam os modelos, associando os algoritmos de forma a minimizar suas desvantagens individuais no modelo final.

Quando apenas um modelo base é treinado em momentos diferentes e posteriormente é realizada a combinação, formando assim um único modelo, é denominado de modelo homogêneo. Se o agrupamento final for formado por modelos diferentes é denominado como um preditor heterogêneo.

Existem variados métodos de aprendizagem por agrupamento, os tipos mais utilizados são *Bagging*, *Boosting* e *Stacking*. Suas principais características são:

- *Bagging*: é feito com preditores homogêneos, cada um de forma independente em relação ao outro, de forma paralela. O algoritmo final é então feito a partir de algum tipo de resultado médio do que foi obtido a partir dos modelos bases;
- *Boosting*: é feito com preditores homogêneos, que são aplicados de forma sequencial (o posterior depende do antecessor) e depois combinados no modelo final;
- *Stacking*: é feito com preditores heterogêneos, treinando-os em paralelo. É então aplicado um modelo na saída dos modelos iniciais (podendo incluir ou não as *features* utilizadas para treiná-los).

O método de aprendizagem por agrupamento escolhido para o desenvolvimento desta pesquisa foi o *Stacking*, pois ele é criado a partir de modelos separados aproveitando as *features* utilizadas para treiná-los, que serão usadas em um modelo único.

O modelo proposto neste trabalho foi implementado combinando os modelos de três arquiteturas diferentes, aproveitando as *features* utilizadas durante seus respectivos treinamentos. Os submodelos foram incorporados em um modelo maior com várias cabeças que pode aprender a combinar melhor as previsões de cada submodelo de entrada.

2.4 Classificação de imagens

Segundo Schwartz e Pedrini (2003) a análise de imagens é uma das tarefas essenciais envolvidas em um sistema de classificação. Seu propósito é extrair informações suficientes para

identificar diferentes regiões de interesse. Geralmente, o processo de classificação é fundado nos níveis de cinza ou cores, forma e textura. Várias aplicações utilizam técnicas de classificação para reconhecimento de objetos em imagens, como também análise de imagens multiespectrais e interpretação de fotografias aéreas.

A classificação de imagens é sobretudo o método de se isolar imagens em grupos anteriormente definidos, onde imagens relacionadas a um mesmo grupo possuem afinidades significativas, assim como também possuem distinções significativas com imagens dos demais grupos. A definição das regras de classificação de imagens consiste na extração de características das mesmas.

Para que se torne viável a construção de um classificador automático Silva, Julia e Barcelos (2012, p.3) declaram que “as regras de classificação das imagens têm que ser expressas numericamente, oferecendo, desta forma, que uma técnica desenvolvida ou adaptada a reconhecer padrões dentro destas regras”. Prontamente, para o processamento de imagens, são selecionadas regras baseadas na intensidade dos pixels para se classificar objetos ou regiões das imagens.

2.4.1 Imagens tomográficas

A tomografia computadorizada (TC) é um exame de diagnóstico por imagem, que consiste em uma imagem que representa uma secção ou "fatia" do corpo. É obtida através do processamento por computador recolhida após expor o corpo a uma sucessão de raios X. Seu método principal é analisar a atenuação de um feixe de raios X durante seu trajeto através de um segmento do corpo (Amaro, 2001).

Entre as características das imagens tomográficas destacam-se os pixels, a matriz, o campo de visão, a escala de cinza e as janelas. O pixel é o menor ponto que pode ser obtido em uma imagem. Quanto maior o número de pixels em uma matriz melhor é a sua resolução espacial, o que permite uma melhor diferenciação espacial entre as estruturas.

2.4.2 Imagens funcionais

A imagem funcional é o estado da arte no diagnóstico médico por imagem. Assim, foi mencionado que “a Ressonância Magnética deu seu enorme avanço nessa direção por meio das técnicas de obtenção ultrarrápidas, que possibilitaram medir as alterações no nível de utilização de oxigênio decorrentes do efeito BOLD” (Tinois, 2005, p.1). Nessa linha, existem dois amplos

métodos de imagem que vêm se evidenciando: o PET (*Positron Emission Tomography*) dinâmico e a fMRI (*Functional Magnetic Resonance Imaging*). A primeira advém da Medicina Nuclear e a segunda da Ressonância Magnética. Para a primeira, foi desenvolvido um dispositivo destinado à obtenção das imagens e, para a segunda, técnicas de hardware e de software, aproveitando a mesma base do dispositivo (Queirós, 2011). As imagens funcionais vêm sendo empregadas principalmente em pesquisas sobre o funcionamento cerebral, sendo apto até mesmo de disponibilizar informações para o planejamento cirúrgico.

2.5 Ferramentas e técnicas

Esta seção mostra as ferramentas, frameworks e técnicas que foram utilizadas durante o desenvolvimento dos experimentos.

2.5.1 TensorFlow

TensorFlow é um sistema de aprendizado de máquina que opera em grande escala e em ambientes heterogêneos. O TensorFlow usa grafos de fluxo de dados para representar computação, estado compartilhado e as operações que alteram esse estado. Ele mapeia os nós de um grafo de fluxo de dados em muitas máquinas em um cluster e, dentro de uma máquina, em vários dispositivos computacionais, incluindo CPUs multicore, GPUs de uso geral e ASICs personalizados, conhecidos como Tensor Processing Units (TPUs).

Essa arquitetura oferece flexibilidade ao desenvolvedor de aplicativos: enquanto nos designs de "servidor de parâmetros" anteriores o gerenciamento de estado compartilhado é integrado ao sistema, o TensorFlow permite que os desenvolvedores experimentem novas otimizações e algoritmos de treinamento. Além disso, o TensorFlow é compatível com uma variedade de aplicativos, com foco no treinamento e inferência em redes neurais profundas (Abadi, 2016).

2.5.2 Keras

É a API de alto nível do TensorFlow para criar e treinar modelos de aprendizado profundo. Ela é usada para prototipagem rápida, pesquisa de ponta e produção, com as seguintes vantagens: Dispõe de uma interface simples e consistente otimizada para os casos de uso

comuns e os modelos modulares e compostos do Keras são feitos conectando elementos configuráveis e com poucas restrições (Keras, 2020).

2.5.3 NVIDIA cuDNN

É uma biblioteca de primitivos acelerada por GPU para redes neurais profundas. O cuDNN fornece implementações altamente ajustadas para rotinas padrão, como camadas de convolução forward e backward, pooling, normalização e ativação. cuDNN acelera estruturas de aprendizado profundo amplamente usadas, incluindo Caffe2, Keras, PyTorch e TensorFlow (Chetlur, 2014).

2.5.4 Transfer learning

Transfer learning ou aprendizagem por transferência é a reutilização de um modelo pré-treinado em um novo problema. Os modelos utilizados foram treinados com pesos do ImageNet que possui milhões de imagens naturais em diversas categorias, o treinamento de uma massa de dados tão grande é muito custoso, levam muitos dias em gpus avançadas, além do gasto de energia e, conseqüentemente dinheiro.

O objetivo da Aprendizagem por Transferência é aproveitar todo esse conhecimento que foi gerado usando essas imagens naturais, seja nas camadas mais iniciais, como o reconhecimento de bordas, curvas e cores, quanto nas camadas convolucionais mais internas onde aprendem a textura, liso. Já as camadas mais especializadas não são interessantes para o aprendizado, então somente são utilizadas as camadas iniciais que reconhecem detalhes básicos que teoricamente toda imagem possui.

2.5.5 Callbacks

Um Callback é um objeto que pode realizar ações em vários estágios de treinamento (por exemplo, no início ou no final de uma época, antes ou depois de um único lote, etc.) (Keras, 2020). Os callbacks podem ser utilizados para:

- Gravar registros do TensorBoard após cada lote de treinamento para monitorar suas métricas;
- Salvar periodicamente seu modelo em disco;

- Parar mais cedo;
- Obter uma visão dos estados internos e estatísticas de um modelo durante o treinamento.

3 Definição dos modelos

Os modelos foram criados usando “*include_top = false*” para tirar o pedaço específico das imagens naturais ao iniciar com os pesos do ImageNet. Logo em seguida, é feito um *pooling* para reduzir o tamanho e foram adicionadas algumas camadas densas. Nas camadas densas foi utilizado o “*relu*” como ativação e “*he_normal*” como peso do kernel. Também foi utilizado *dropout* com 0.3 em cada camada densa e por último *softmax* para poder separar as classes.

Foi utilizado Adam com *learning rate* de 0.002, a função de *loss* como *categorical_crossentropy* e *accuracy* como métrica. Logo em seguida, o modelo foi executado usando validação cruzada com $k = 5$ e *fit_generator* com 50 épocas para cada uma das arquiteturas.

Também foram utilizados os seguintes *Callbacks* para realizar ações em vários estágios de treinamento:

- ModelCheckpoint: para salvar o modelo que tiver o melhor *loss* durante o treinamento;
- EarlyStop para interromper o treinamento caso a rede pare de aprender;
- ReduceLROnPlateau para diminuir a taxa de aprendizagem se o *loss* da validação não se alterar.

3.1 Definição do modelo EnsembleDVX

A média do modelo é uma técnica de agrupamento em que vários submodelos contribuem igualmente para uma predição combinada. Entretanto, a média do modelo pode ser melhorada ponderando as contribuições de cada submodelo para a previsão combinada pelo desempenho esperado do submodelo. Isso pode ser estendido ainda mais com o treinamento de um modelo inteiramente novo para aprender como combinar melhor as contribuições de cada submodelo. Esta abordagem é chamada de generalização empilhada (*Stacked Generalization*) ou empilhamento, e pode resultar em um melhor desempenho preditivo do que qualquer modelo único, já que no empilhamento, um algoritmo pega as saídas de submodelos como entrada e tenta aprender como combinar melhor as previsões de entrada para fazer uma melhor previsão de saída.

A generalização empilhada funciona deduzindo os vieses do generalizador(es) em relação a um conjunto de aprendizagem fornecido. Esta dedução prossegue generalizando em um segundo espaço cujas entradas são (por exemplo) as suposições dos generalizadores originais quando ensinados com parte do conjunto de aprendizagem e tentando adivinhar o resto, e cuja saída é (por exemplo) a suposição correta (Wolpert, 1992).

O modelo EnsembleDVX foi implementado combinando os modelos de três arquiteturas diferentes, são elas: DenseNet169, VGG16 e Xception. As sub-redes foram incorporadas em uma rede neural maior com várias cabeças que, então, pode aprender a combinar melhor as previsões de cada submodelo de entrada. Permitindo assim, que o conjunto de empilhamento seja tratado como um único modelo grande.

O benefício dessa abordagem é que os resultados dos submodelos são fornecidos diretamente ao modelo único. Além disso, também é possível atualizar os pesos dos submodelos em conjunto com o modelo único, se isso for desejável.

O procedimento foi realizado utilizando a interface funcional do Keras, onde os modelos são carregados como uma lista, um modelo de agrupamento por empilhamento maior então é definido, onde cada um dos modelos carregados é usado como um cabeçote de entrada separados para o modelo. Isso requer que todas as camadas em cada um dos modelos carregados sejam marcadas como não treináveis, para que os pesos não possam ser atualizados quando o novo modelo maior estiver sendo treinado.

O Keras também requer que cada camada tenha um nome exclusivo, portanto, os nomes de cada camada em cada um dos modelos carregados foram atualizados para indicar qual membro do agrupamento eles pertencem.

Uma vez que os submodelos foram preparados, foi definido o modelo de agrupamento por empilhamento. A camada de entrada para cada um dos submodelos foi usada como um cabeçalho de entrada separado para o novo modelo. Isso significa que k cópias de quaisquer dados de entrada foram fornecidas ao modelo, onde k é o número de modelos de entrada, neste caso, 3.

As saídas de cada um dos modelos foram mescladas usando uma mesclagem de concatenação simples, onde um único vetor foi criado a partir das duas probabilidades de classe previstas por cada um dos 3 modelos.

Em seguida, foi definido uma camada oculta utilizando o “*relu*” como ativação para interpretar essa entrada para o modelo e uma camada de saída com “*softmax*” para poder separar as classes que fará sua própria previsão probabilística. Então foi criado uma função que retorna

um modelo de rede neural de generalização empilhado dada uma lista de submodelos treinados. Um gráfico da rede criada pode ser observado na figura 5.

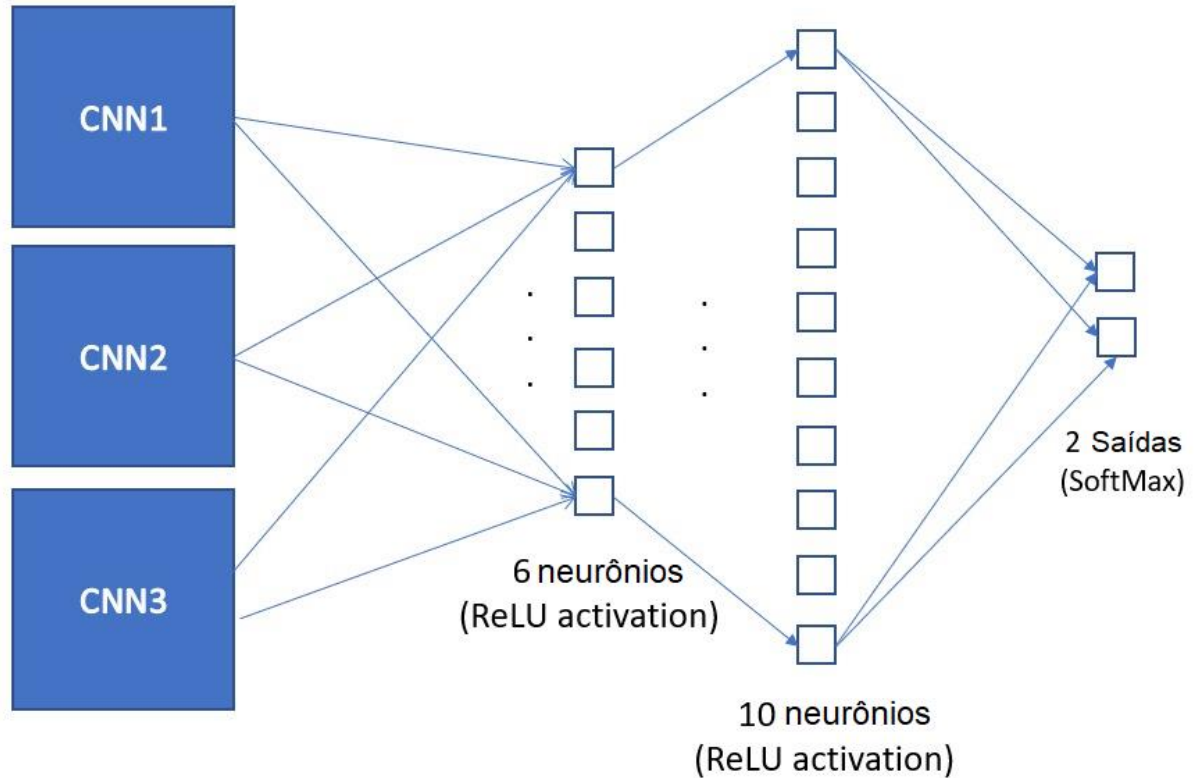


Figura 5 – Gráfico EnsembleDVX (Lúcio Silva, 2021).

Como os submodelos não são treináveis, seus pesos não foram atualizados durante o treinamento e apenas os pesos da nova camada oculta e de saída foram atualizados. O modelo foi executado usando validação cruzada com $k = 5$ e `fit_generator` com 50 épocas.

Também foram utilizados os seguintes Callbacks para realizar ações em vários estágios de treinamento:

- `ModelCheckpoint`: para salvar o modelo que tiver o melhor *loss* durante o treinamento;
- `EarlyStop` para interromper o treinamento caso a rede pare de aprender;
- `ReduceLROnPlateau` para diminuir a taxa de aprendizagem se o *loss* da validação não se alterar.

4 Experimentos

Esta seção detalha como os experimentos foram realizados, qual equipamento usado durante o processo e como os dados foram obtidos e tratados para a realização do treinamento das redes neurais, além das métricas utilizadas para avaliação das arquiteturas.

4.1 Ambiente

Todos os experimentos foram realizados usando uma máquina com as seguintes configurações:

- CPU Intel Core i7 3770 @ 3.40GHz;
- Memória de 8GB Dual-Channel DDR3 @ 798MHz (10-10-10-30);
- GPU NVIDIA GeForce GTX 960 4GB (ZOTAC International);
- SSD 240GB KINGSTON (SATA-3).

4.2 Conjunto de dados

Foi utilizando um dataset do Kaggle denominado de SARS-COV-2 Ct-Scan Dataset disponibilizado por (Soares et al., 2020), que uniu os pesquisadores aos dados do estudo e esse dataset foi utilizado para o desenvolvimento desta pesquisa. O Dataset é composto por 2477 imagens de tomografia computadorizada, que é utilizada para detecção de doenças pulmonares, sendo 1250 de pacientes que testaram positivo para a COVID-19 e 1227 que tiveram diagnóstico negativo. As figuras 6 e 7 exibem uma amostra do dataset com imagens de pessoas com diagnósticos positivos e negativos respectivamente.

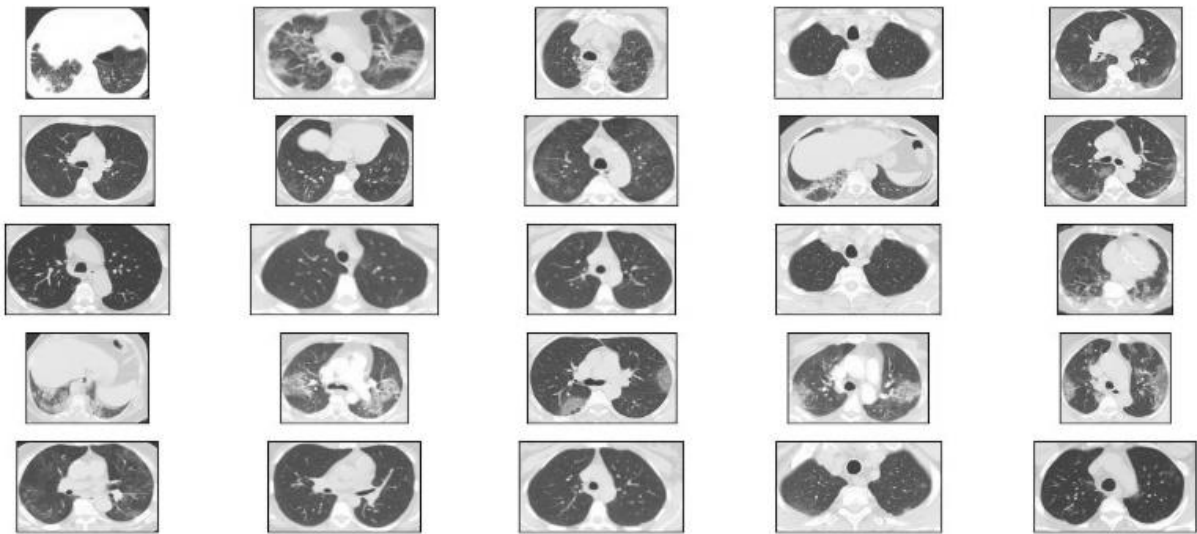


Figura 6 – Imagens Covid-19 padronizadas (Soares et al., 2020).

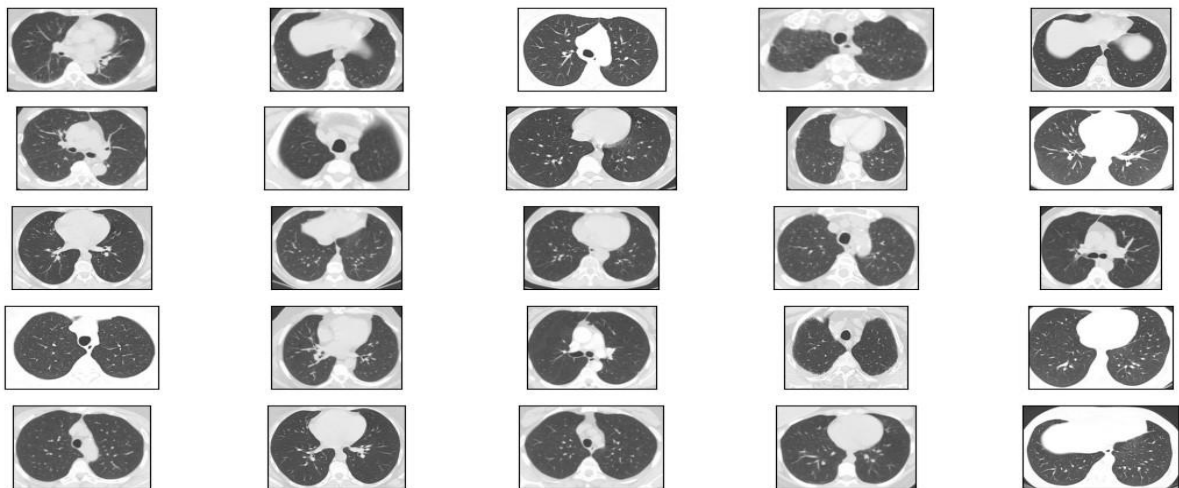


Figura 7 – Imagens Non Covid-19 padronizadas (Soares et al., 2020).

O dataset possui imagens com diferentes resoluções, devido a isso, todas as imagens foram convertidas para a resolução 128x128. Além disso, foi realizado a padronização das imagens e separação dos conjuntos de validação usando **ImageDataGenerator**, essa técnica também faz *Image Augmentation* que vai gerando novas imagens enquanto separa e normaliza. Foi realizado também as seguintes operações a fim de evitar o *overfitting* da rede neural: *rotation range = 10*, *width shift range = 0.2*, *height shift ranged = 0.2*, *zoom range = 0.1*, *horizontal flip = True*, e *vertical flip = True*.

4.3 Métricas de avaliação

Inicialmente, foi definido o significado dos verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. Os verdadeiros positivos e os verdadeiros negativos são classificações corretas, ou seja, classificando COVID-19 e Não-COVID-19 corretamente. Em contraste, falso positivo e falso negativo são classificações erradas. Assim, foram definidas as métricas.

A Equação 2 apresenta a primeira métrica denominada precision. Esta métrica é a proporção entre verdadeiros positivos e verdadeiros positivos mais falsos positivos. Assim, uma baixa precision indica que o número de classificações corretas é muito baixo, ou o número de falsos positivos é alto.

$$P = \frac{VP}{VP + FP} \quad (2)$$

A próxima métrica é a accuracy, conforme apresentada na Equação 3, que é a porcentagem de classificações corretas. Uma baixa accuracy pode indicar que o número de classificações erradas (falsos positivos e falsos negativos) é alto.

$$A = \frac{VP + VN}{VP + FP + VN + FN} \quad (3)$$

A recall apresentado na Equação 4 é a razão entre os verdadeiros positivos e verdadeiros positivos mais falsos positivos. Esta métrica indica que o algoritmo está funcionando bem em classificar verdadeiros positivos. Porém, se essa métrica for baixa, pode significar que um grande número de erros de classificação está acontecendo. Essa métrica é essencial para minimizar o número de falsos negativos, que podem produzir o pior cenário.

$$R = \frac{VP}{VP + FN} \quad (4)$$

Por último, a F1 Score apresentada na Equação 5 é a média harmônica entre a precision e a recall. A F1 acaba sendo uma grande figura do desempenho porque a precision leva em consideração falsos positivos e a recall leva em consideração os falsos negativos. Assim, F1 Score dá uma ideia se o algoritmo de classificação fornece muitas classificações incorretas.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (5)$$

4.4 Resultados

Na Tabela 2 é apresentada a média de todas as métricas utilizadas durante o treinamento. Como pode ser observado, a EnsembleDVX apresentada, alcançou a melhor acurácia, precisão, recall e F1_score com 97,7%, 97,7%, 97,8% e 97,7%, respectivamente. Seguido pela DenseNet169 que também alcançou bons resultados, enquanto que a MobileNet V2 apresentou os piores resultados.

Tabela 2 – Resultado das 12 CNNs.

Modelo	Média Acurácia	Média Precisão	Média Recall	Média F1_Score	Média Perda
EnsembleDVX	0.977±0.002	0.977±0.002	0.978±0.002	0.977±0.002	0.077±0.003
DenseNet169	0.963±0.005	0.963±0.005	0.964±0.005	0.964±0.005	0.097±0.007
VGG16	0.951±0.006	0.952±0.004	0.951±0.007	0.951±0.006	0.134±0.018
Xception	0.950±0.008	0.951±0.008	0.951±0.008	0.951±0.008	0.150±0.015
DenseNet201	0.948±0.010	0.949±0.010	0.948±0.011	0.948±0.010	0.138±0.022
DenseNet121	0.946±0.007	0.946±0.008	0.947±0.007	0.947±0.007	0.155±0.005
VGG19	0.946±0.012	0.947±0.011	0.946±0.013	0.946±0.012	0.134±0.019
ResNet50	0.942±0.009	0.943±0.010	0.942±0.010	0.942±0.009	0.149±0.026
ResNet50V2	0.939±0.010	0.940±0.010	0.940±0.010	0.939±0.010	0.163±0.026
InceptionV3	0.939±0.017	0.940±0.015	0.938±0.018	0.939±0.017	0.161±0.035
MobileNet	0.934±0.011	0.935±0.010	0.934±0.012	0.934±0.011	0.165±0.029
MobileNetV2	0.617±0.062	0.634±0.088	0.613±0.075	0.588±0.086	0.649±0.025

Fonte: Lúcio Silva, 2020.

Considerando os resultados alcançados pelas arquiteturas testadas, estes serão detalhados a seguir.

A DenseNet169 apresentou o segundo melhor resultado, a Figura 8 mostra sua matriz de confusão média, na qual podemos observar que a DenseNet169 estava incorreta em apenas 18 casos, no entanto, 12 casos deram falsos positivos para COVID-19. Por outro lado, o fato notável é que apenas 6 casos de COVID-19 foram erroneamente classificados como não-COVID-19. Abaixo, detalhamos os resultados da DenseNet169, mostrando a matriz de confusão e suas curvas de treinamento.

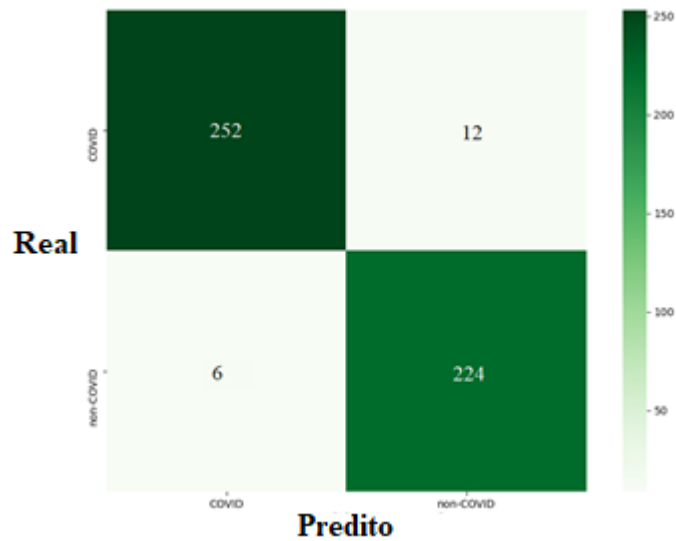


Figura 8 – Matriz de Confusão – DenseNet169 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 03, onde foram executadas com $k = 5$.

Tabela 3 – Folds DenseNet169.

Folds	Acurácia	Perda
Fold 1	0.959	0.102
Fold 2	0.965	0.088
Fold 3	0.969	0.097
Fold 4	0.967	0.107
Fold 5	0.957	0.092

Fonte: Lúcio Silva, 2021.

As Figuras 8.1 a 8.5 mostram a acurácia e a perda. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) na perda. Em outras palavras, as curvas tendem a convergir.

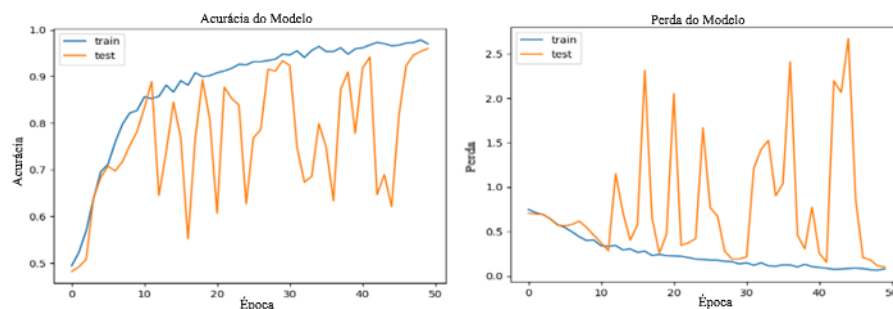


Figura 8.1 – Acurácia e Perda Fold 1 - DenseNet169 (Lúcio Silva, 2021).

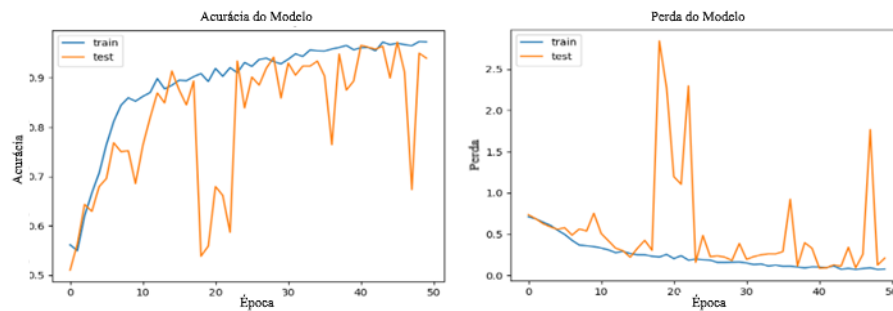


Figura 8.2 – Acurácia e Perda Fold 2 - DenseNet169 (Lúcio Silva, 2021).

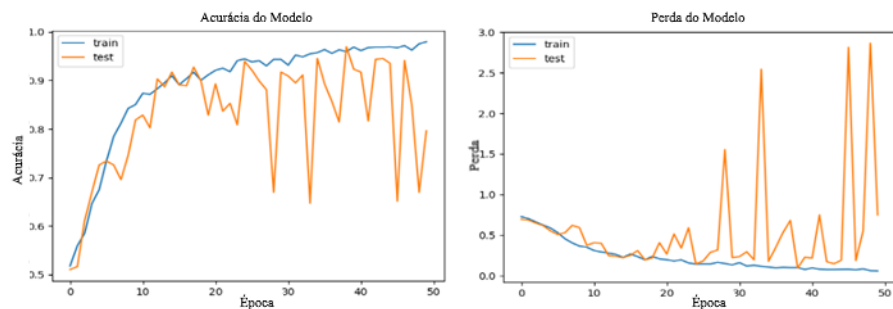


Figura 8.3 – Acurácia e Perda Fold 3 - DenseNet169 (Lúcio Silva, 2021).

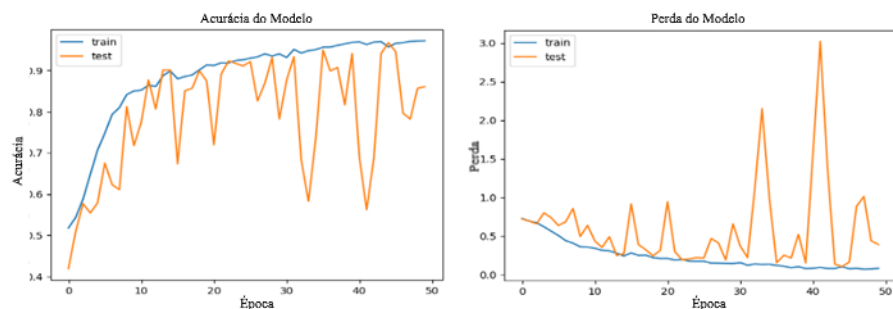


Figura 8.4 – Acurácia e Perda Fold 4 - DenseNet169 (Lúcio Silva, 2021).

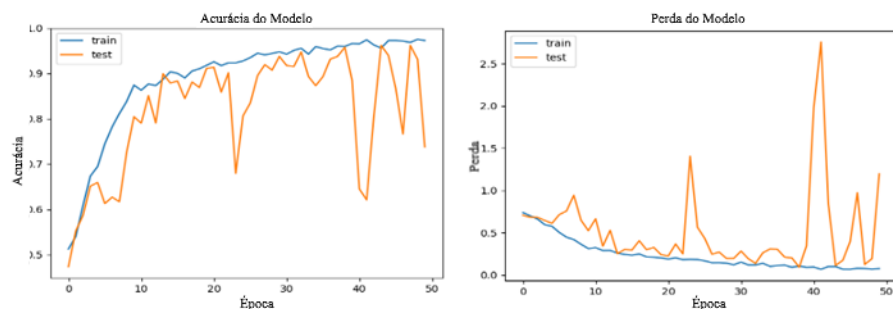


Figura 8.5 – Acurácia e Perda Fold 5 - DenseNet169 (Lúcio Silva, 2021).

A VGG16 também obteve bons resultados. A Figura 10 mostra sua matriz de confusão média, na qual podemos observar que a VGG16 estava incorreta em 24 casos, no entanto, 12 casos deram falso positivo para COVID-19 e 12 casos de COVID-19 foram erroneamente classificados como não-COVID-19. Abaixo, detalhamos os resultados da VGG16, mostrando a matriz de confusão e suas curvas de treinamento.

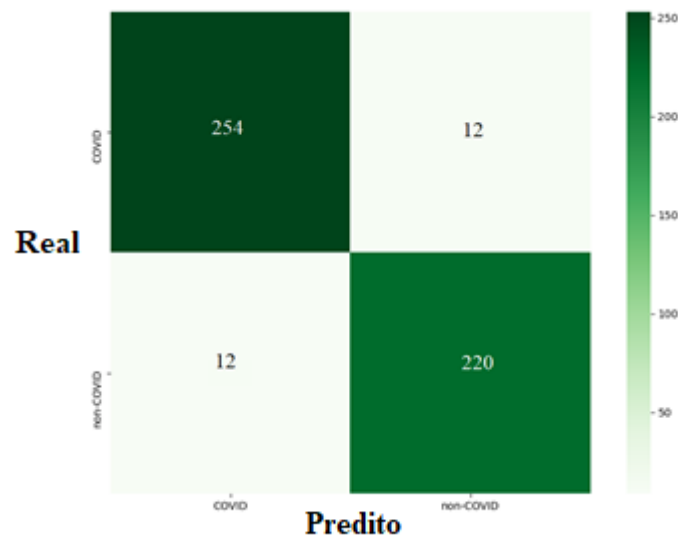


Figura 9 – Matriz de Confusão – VGG16 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 04, onde foram executadas com $k = 5$.

Tabela 4 – Folds VGG16.

Folds	Acurácia	Perda
Fold 1	0.951	0.152
Fold 2	0.951	0.136
Fold 3	0.957	0.108
Fold 4	0.941	0.155
Fold 5	0.955	0.121

Fonte: Lúcio Silva, 2021.

As Figuras 9.1 a 9.5 mostram a acurácia e a perda alcançadas pela VGG16. Como pode ser observado, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda. Em outras palavras, as curvas tendem a convergir.

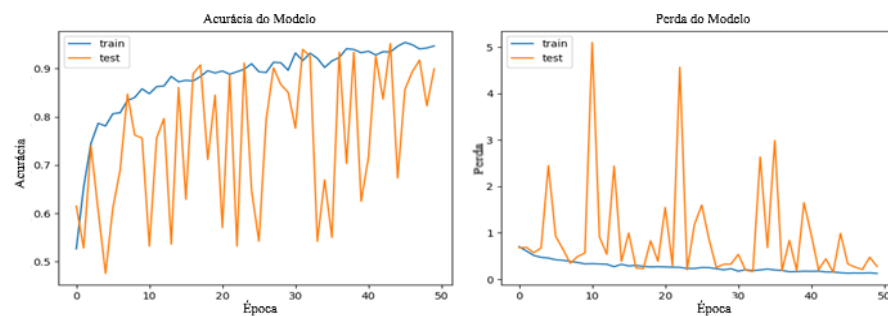


Figura 9.1 – Acurácia e Perda Fold 1 - VGG16 (Lúcio Silva, 2021).

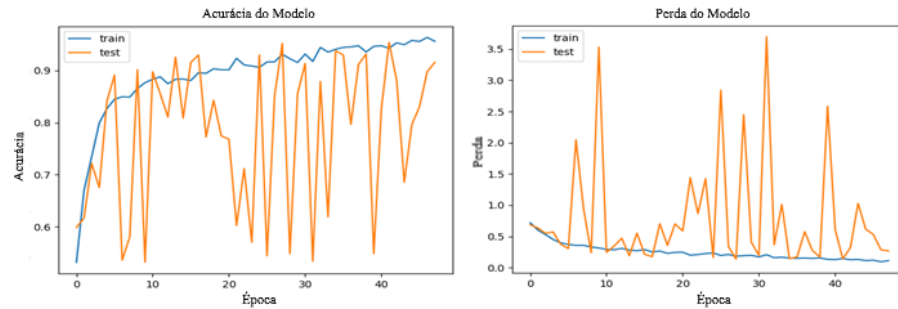


Figura 9.2 – Acurácia e Perda Fold 2 - VGG16 (Lúcio Silva, 2021).

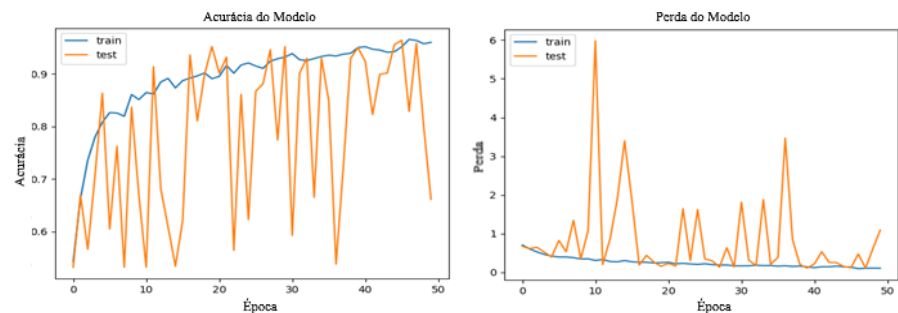


Figura 9.3 – Acurácia e Perda Fold 3 - VGG16 (Lúcio Silva, 2021).

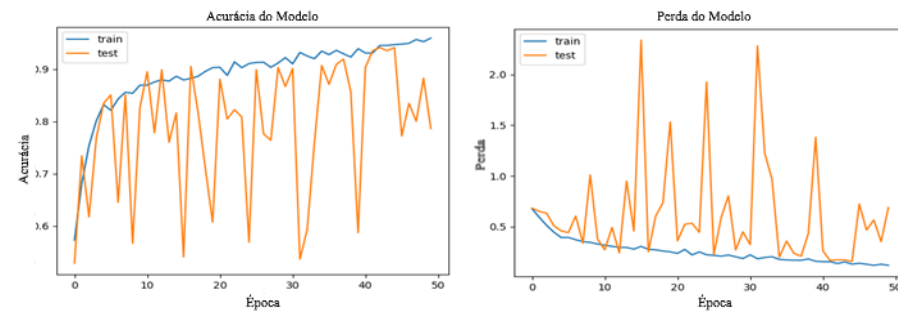


Figura 9.4 – Acurácia e Perda Fold 4 - VGG16 (Lúcio Silva, 2021).

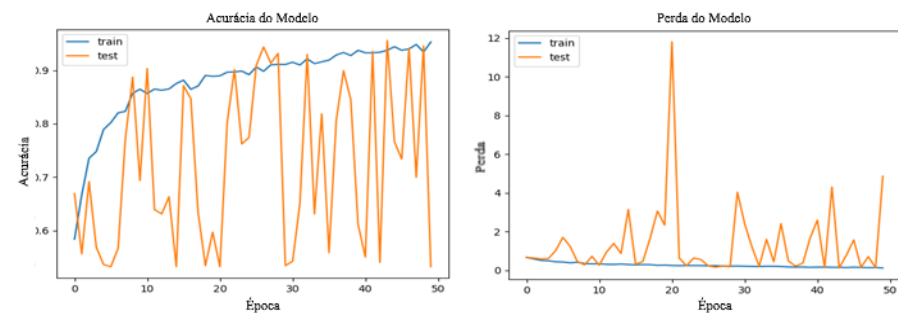


Figura 9.5 – Acurácia e Perda Fold 5 - VGG16 (Lúcio Silva, 2021).

A Xception, que também obteve bons resultados, a Figura 10 mostra sua matriz de confusão média, onde podemos observar que a Xception estava incorreta em 24 casos, entretanto, 14 casos deram falsos positivos para COVID-19. Por outro lado, apenas 10 casos de COVID-19 foram erroneamente classificados como não-COVID-19. Abaixo, detalhamos os resultados da Xception, mostrando a matriz de confusão e suas curvas de treinamento.

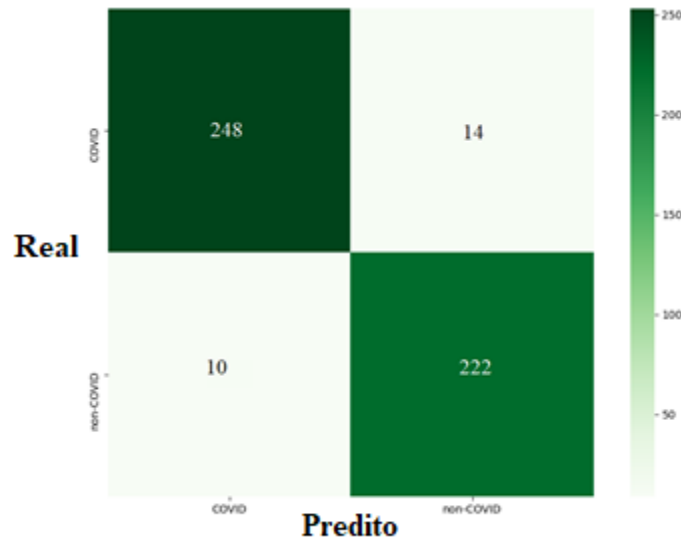


Figura 10 – Matriz de Confusão – Xception (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 05, onde foram executadas com $k = 5$.

Tabela 5 – Folds Xception.

Folds	Acurácia	Perda
Fold 1	0.949	0.171
Fold 2	0.965	0.126
Fold 3	0.951	0.143
Fold 4	0.943	0.152
Fold 5	0.945	0.160

Fonte: Lúcio Silva, 2021.

As Figuras 10.1 a 10.5 apresentam a acurácia e a perda alcançados pela Xception. Como pode ser observado, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda. Em outras palavras, as curvas tendem a convergir.

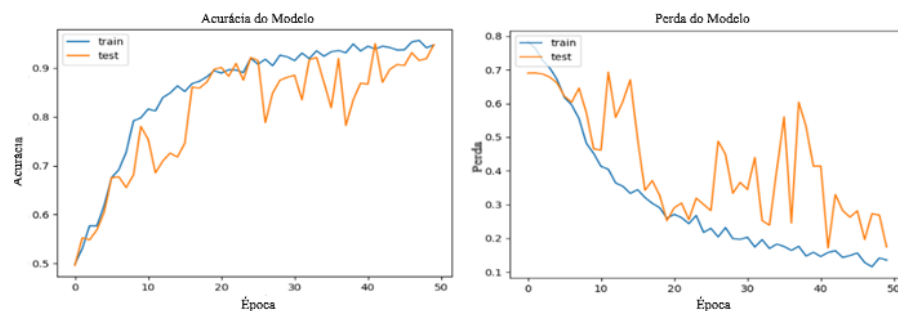


Figura 10.1 – Acurácia e Perda Fold 1 – Xception (Lúcio Silva, 2021).

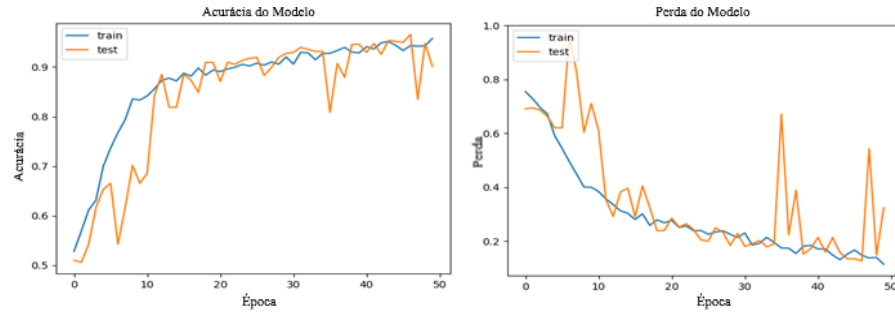


Figura 10.2 – Acurácia e Perda Fold 2 – Xception (Lúcio Silva, 2021).

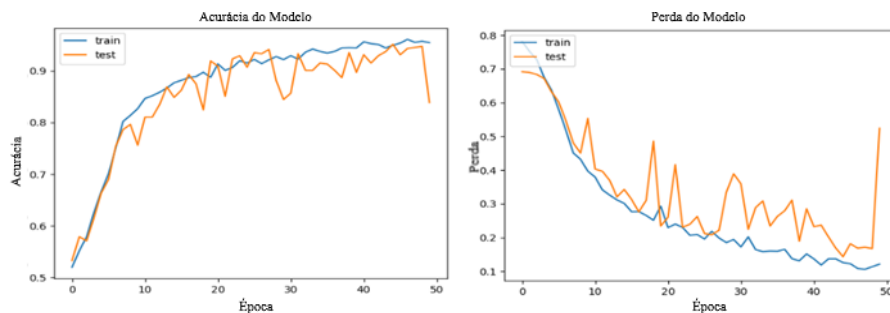


Figura 10.3 – Acurácia e Perda Fold 3 – Xception (Lúcio Silva, 2021).

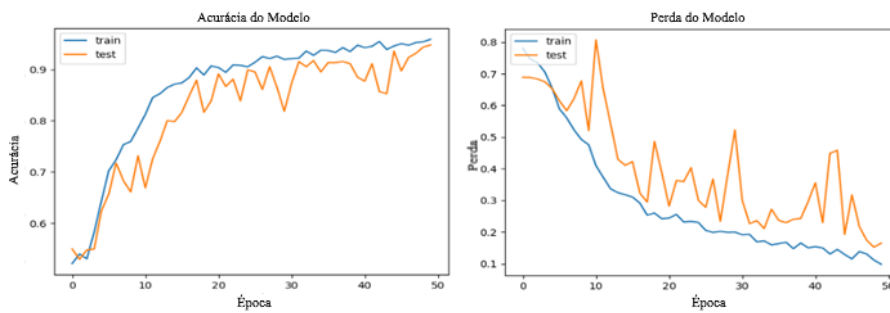


Figura 10.4 – Acurácia e Perda Fold 4 – Xception (Lúcio Silva, 2021).

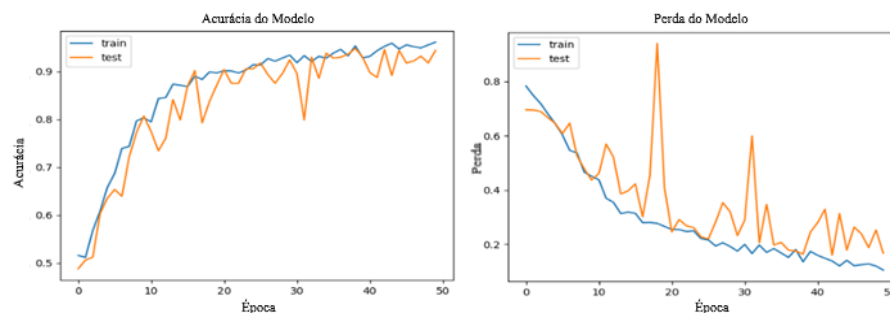


Figura 10.5 – Acurácia e Perda Fold 5 – Xception (Lúcio Silva, 2021).

Os resultados alcançados pela DenseNet201 podem ser observados na Figura 11, que mostra sua matriz de confusão média, na qual podemos observar que a DenseNet201 estava incorreta em 25 casos, 13 casos deram falsos positivos para COVID-19 e 12 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da DenseNet201, mostrando a matriz de confusão e suas curvas de treinamento.

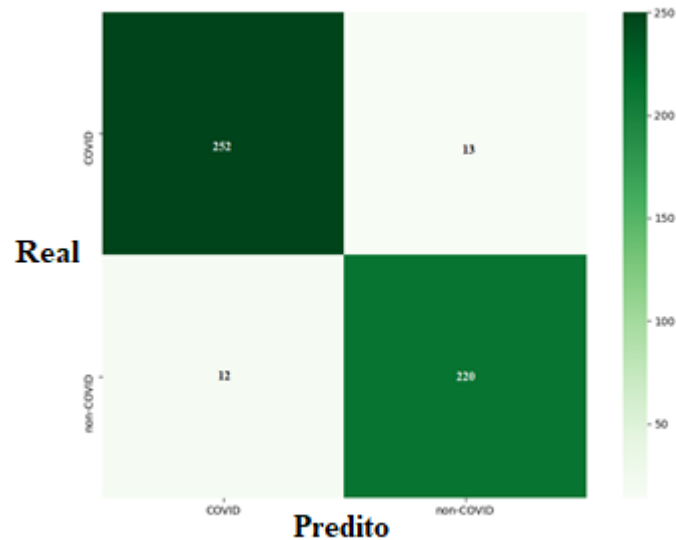


Figura 11 – Matriz de Confusão – DenseNet201 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 06, onde foram executadas com $k = 5$.

Tabela 6 – Folds DenseNet201.

Folds	Acurácia	Perda
Fold 1	0.935	0.163
Fold 2	0.943	0.158
Fold 3	0.943	0.144
Fold 4	0.961	0.119
Fold 5	0.959	0.106

Fonte: Lúcio Silva, 2020.

As Figuras 11.1 a 11.5 mostram a acurácia e a perda alcançadas pela DenseNet201. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

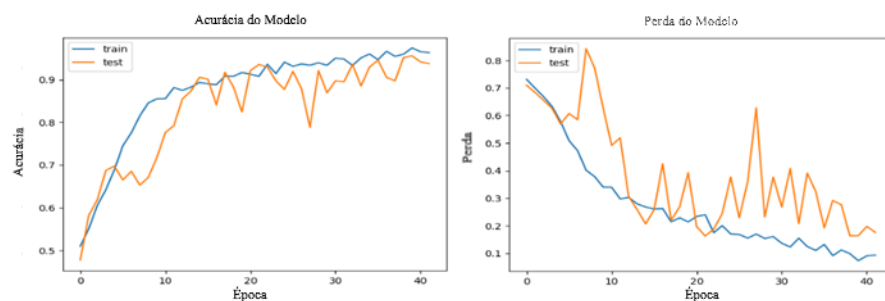


Figura 11.1 – Acurácia e Perda Fold 1 - DenseNet201 (Lúcio Silva, 2021).

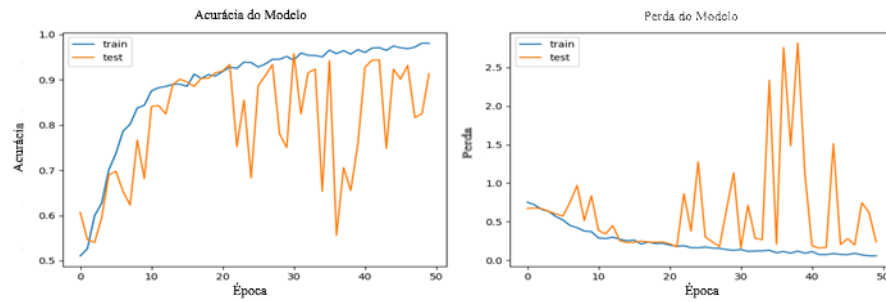


Figura 11.2 – Acurácia e Perda Fold 2 - DenseNet201 (Lúcio Silva, 2021).

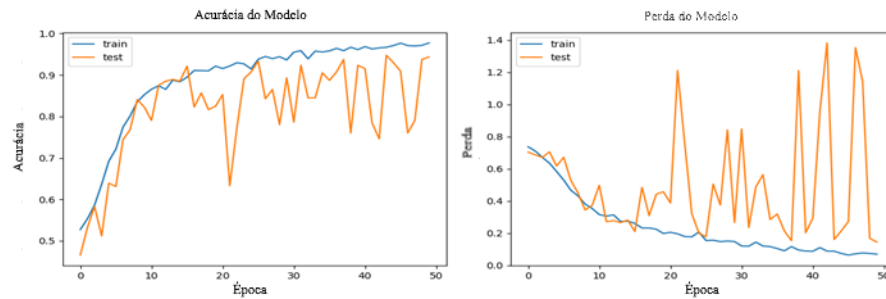


Figura 11.3 – Acurácia e Perda Fold 3 - DenseNet201 (Lúcio Silva, 2021).

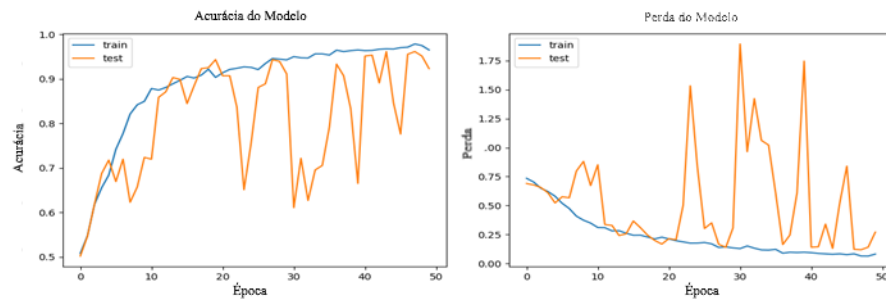


Figura 11.4 – Acurácia e Perda Fold 4 - DenseNet201 (Lúcio Silva, 2021).

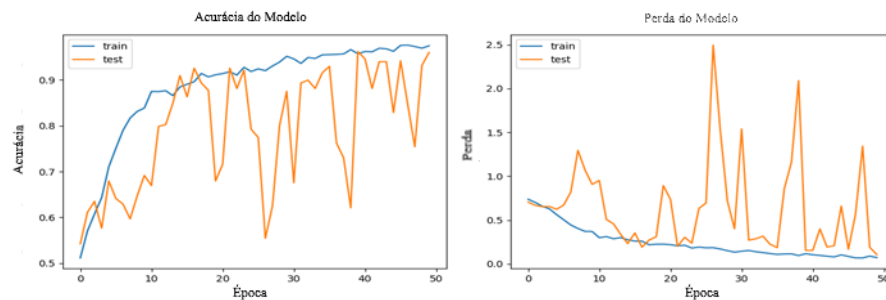


Figura 11.5 – Acurácia e Perda Fold 5 - DenseNet201 (Lúcio Silva, 2021).

Os resultados alcançados pela DenseNet121 são detalhados na Figura 12, que mostra sua matriz de confusão média, na qual podemos observar que a DenseNet121 estava incorreta em 26 casos, 15 casos deram falsos positivos para COVID-19 e 11 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da DenseNet121, mostrando a matriz de confusão e suas curvas de treinamento.

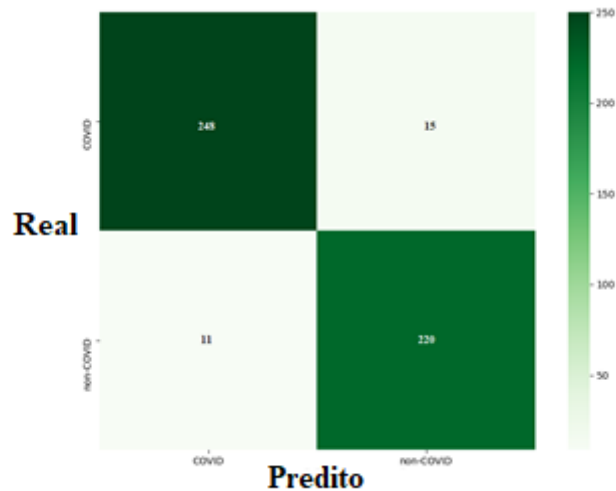


Figura 12 – Matriz de Confusão – DenseNet121 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 07, onde foram executadas com $k = 5$.

Tabela 7 – Folds DenseNet121.

Folds	Acurácia	Perda
Fold 1	0.953	0.161
Fold 2	0.957	0.153
Fold 3	0.943	0.148
Fold 4	0.943	0.161
Fold 5	0.937	0.152

Fonte: Lúcio Silva, 2021.

As Figuras 12.1 a 12.5 mostram a acurácia e a perda alcançadas pela DenseNet121. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

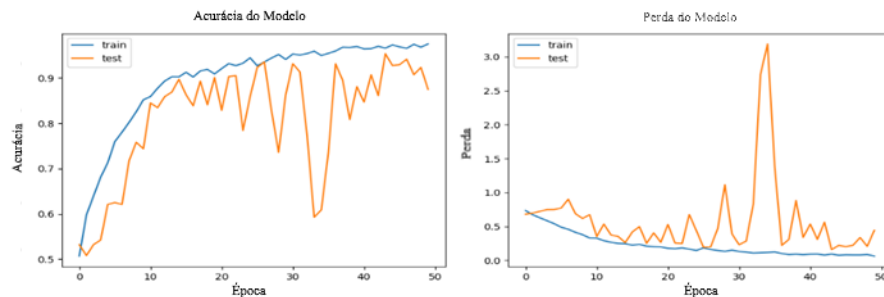


Figura 12.1 – Acurácia e Perda Fold 1 - DenseNet121 (Lúcio Silva, 2021).

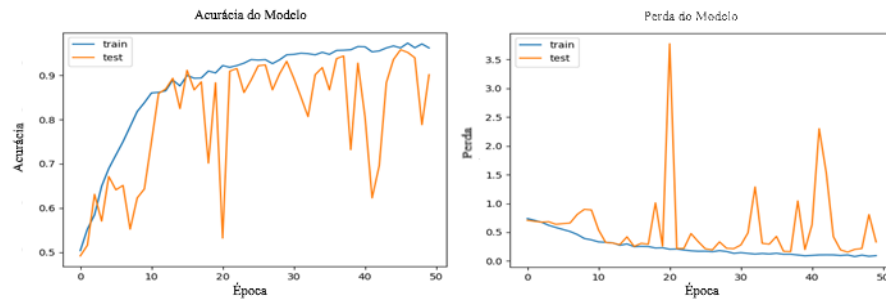


Figura 12.2 – Acurácia e Perda Fold 2 - DenseNet121 (Lúcio Silva, 2021).

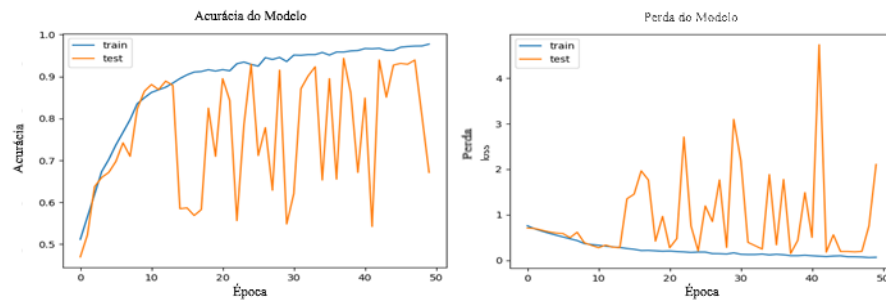


Figura 12.3 – Acurácia e Perda Fold 3 - DenseNet121 (Lúcio Silva, 2021).

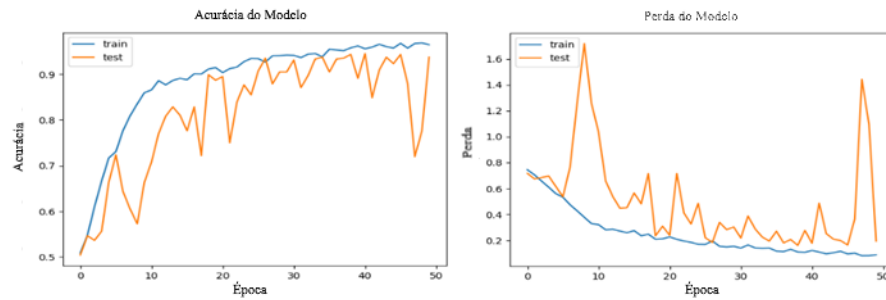


Figura 12.4 – Acurácia e Perda Fold 4 - DenseNet121 (Lúcio Silva, 2021).

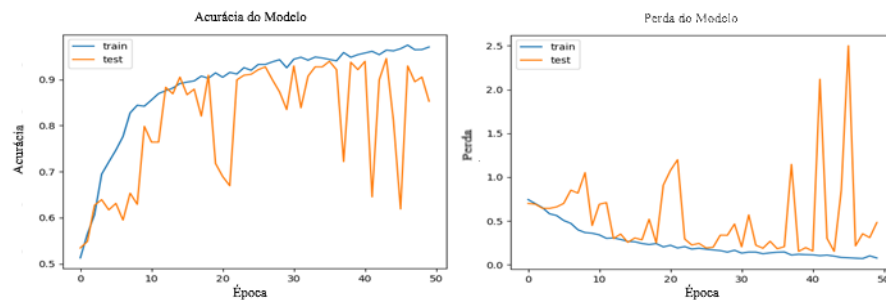


Figura 12.5 – Acurácia e Perda Fold 5 - DenseNet121 (Lúcio Silva, 2021).

Os resultados alcançados pela VGG19 são detalhados na Figura 13, que mostra sua matriz de confusão média, na qual podemos observar que a VGG19 estava incorreta em 27 casos, porém, 12 casos deram falsos positivos para COVID-19 e 15 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da VGG19, mostrando a matriz de confusão e suas curvas de treinamento.

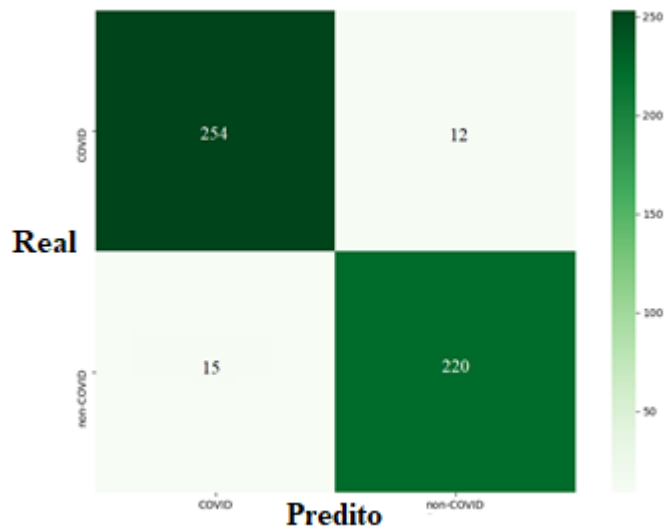


Figura 13 – Matriz de Confusão – VGG19 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 08, onde foram executadas com $k = 5$.

Tabela 8 – Folds VGG19.

Folds	Acurácia	Perda
Fold 1	0.967	0.109
Fold 2	0.937	0.163
Fold 3	0.947	0.135
Fold 4	0.933	0.144
Fold 5	0.947	0.121

Fonte: Lúcio Silva, 2021.

As Figuras 13.1 a 13.5 mostram a acurácia e a perda alcançadas pela VGG19. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

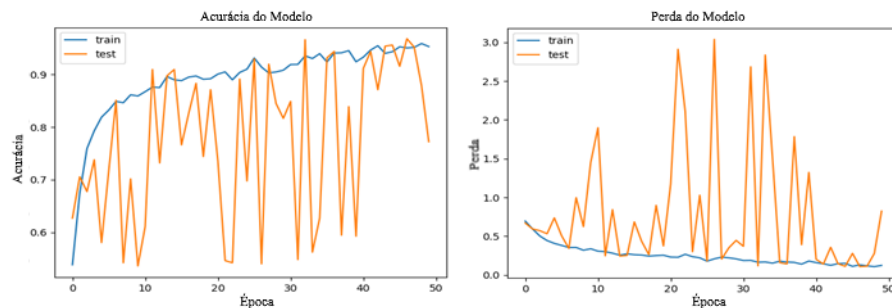


Figura 13.1 – Acurácia e Perda Fold 1 - VGG19 (Lúcio Silva, 2021).

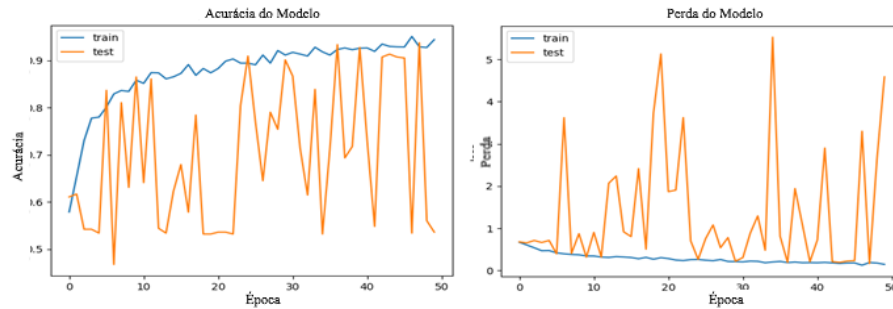


Figura 13.2 – Acurácia e Perda Fold 2 - VGG19 (Lúcio Silva, 2021).

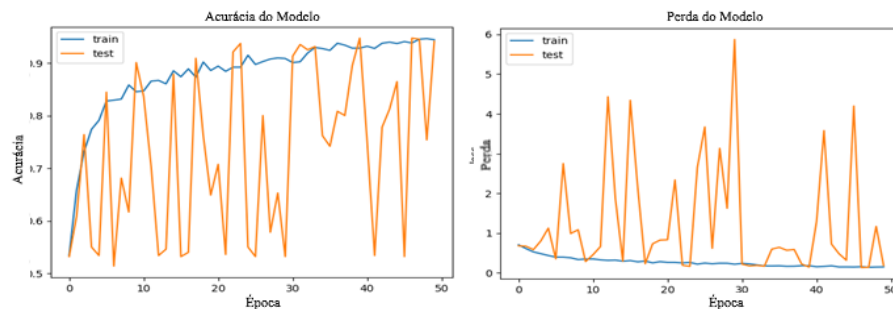


Figura 13.3 – Acurácia e Perda Fold 3 - VGG19 (Lúcio Silva, 2021).

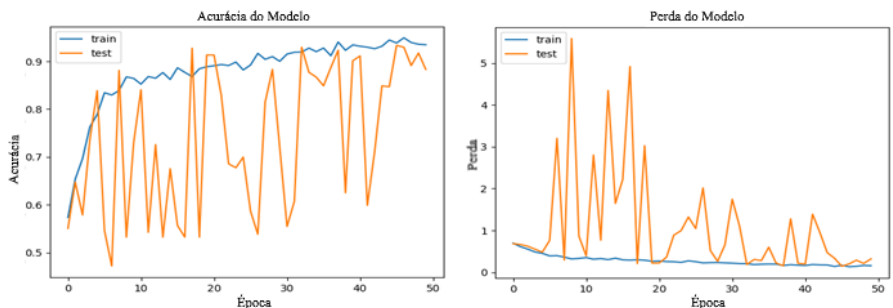


Figura 13.4 – Acurácia e Perda Fold 4 - VGG19 (Lúcio Silva, 2021).

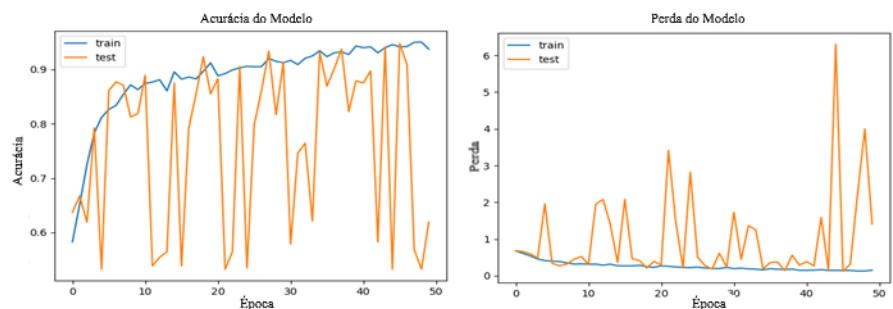


Figura 13.5 – Acurácia e Perda Fold 5 - VGG19 (Lúcio Silva, 2021).

Os resultados alcançados pela ResNet50 podem ser observados na Figura 14, que mostra sua matriz de confusão média, na qual podemos observar que a ResNet50 estava incorreta em 28 casos, 14 casos deram falsos positivos para COVID-19 e 14 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da ResNet50, mostrando a matriz de confusão e suas curvas de treinamento.

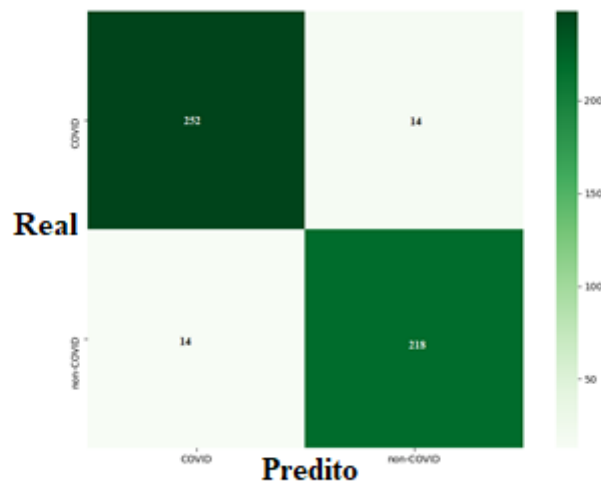


Figura 14 – Matriz de Confusão – ResNet50 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 09, onde foram executadas com $k = 5$.

Tabela 9 – Folds ResNet50.

Folds	Acurácia	Perda
Fold 1	0.941	0.165
Fold 2	0.947	0.136
Fold 3	0.959	0.108
Fold 4	0.931	0.170
Fold 5	0.935	0.166

Fonte: Lúcio Silva, 2021.

As Figuras 15.1 a 15.5 mostram a acurácia e a perda alcançadas pela ResNet50. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

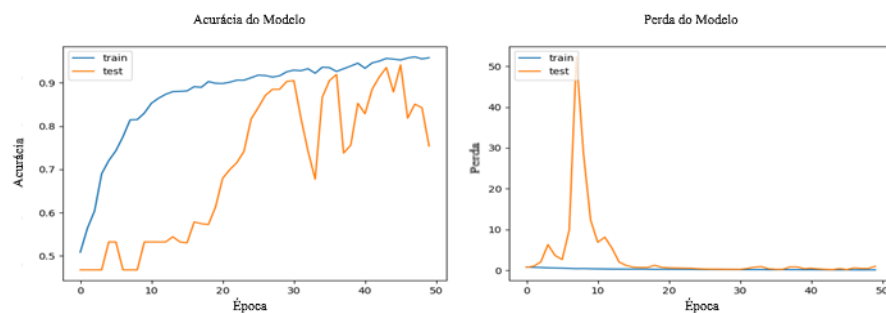


Figura 14.1 – Acurácia e Perda Fold 1 - ResNet50 (Lúcio Silva, 2021).

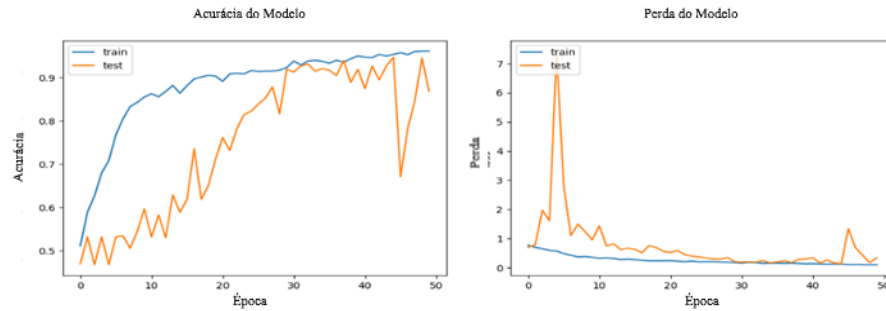


Figura 14.2 – Acurácia e Perda Fold 2 - ResNet50 (Lúcio Silva, 2021).

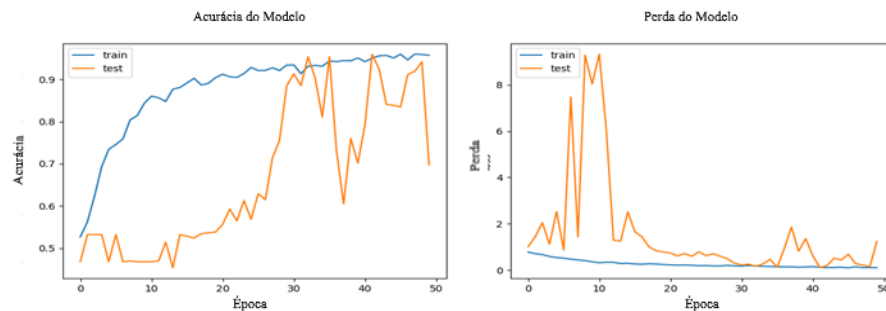


Figura 14.3 – Acurácia e Perda Fold 3 - ResNet50 (Lúcio Silva, 2021).

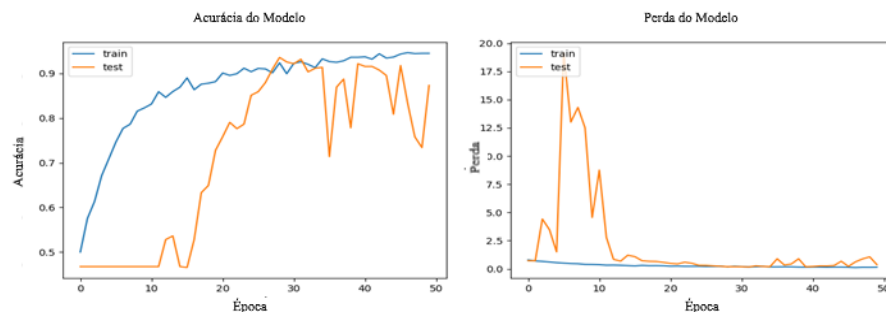


Figura 14.4 – Acurácia e Perda Fold 4 - ResNet50 (Lúcio Silva, 2021).

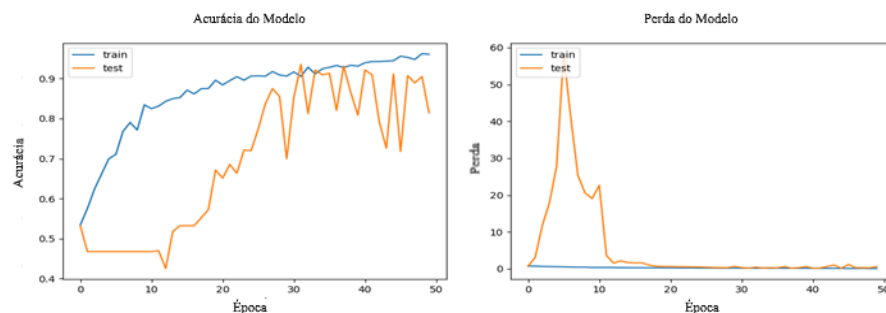


Figura 14.5 – Acurácia e Perda Fold 5 - ResNet50 (Lúcio Silva, 2021).

Os resultados alcançados pela ResNet50V2 podem ser observados na Figura 15, que mostra sua matriz de confusão média, na qual podemos observar que a ResNet50V2 estava incorreta em 30 casos, 17 casos deram falsos positivos para COVID-19 e 13 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da ResNet50V2, mostrando a matriz de confusão e suas curvas de treinamento.

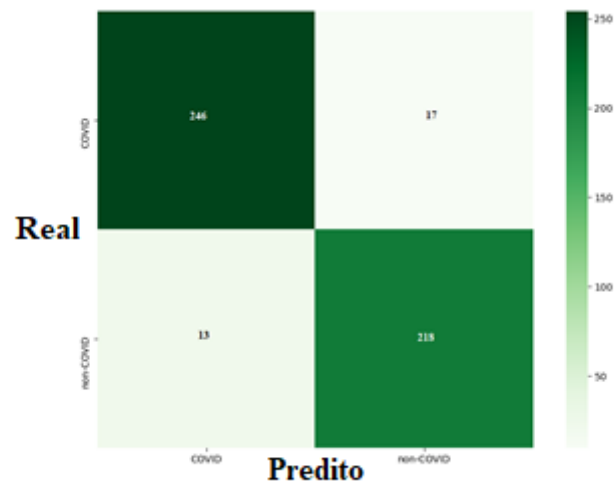


Figura 15 – Matriz de Confusão – ResNet50V2 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 10, onde foram executadas com $k = 5$.

Tabela 10 – Folds ResNet50V2.

Folds	Acurácia	Perda
Fold 1	0.933	0.178
Fold 2	0.949	0.134
Fold 3	0.943	0.132
Fold 4	0.949	0.195
Fold 5	0.923	0.176

Fonte: Lúcio Silva, 2021.

As Figuras 15.1 a 15.5 mostram a acurácia e a perda alcançados pela ResNet50V2. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

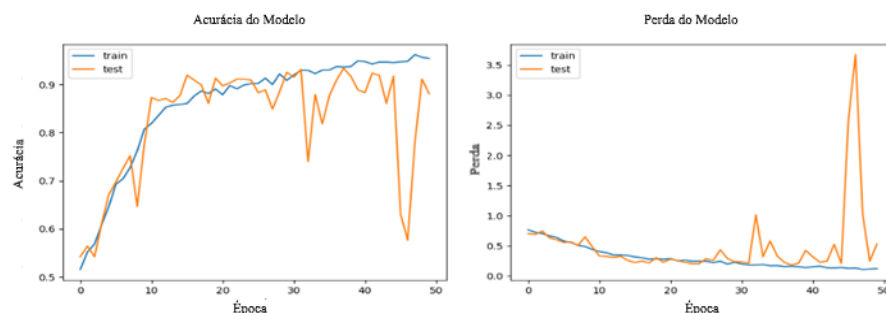


Figura 15.1 – Acurácia e Perda Fold 1 - ResNet50V2 (Lúcio Silva, 2021).

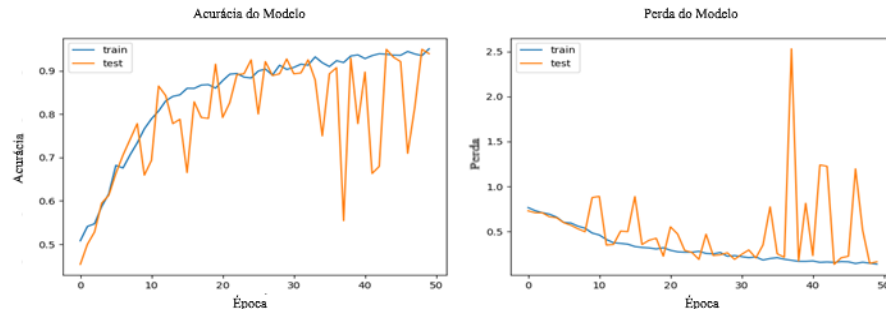


Figura 15.2 – Acurácia e Perda Fold 2 - ResNet50V2 (Lúcio Silva, 2021).

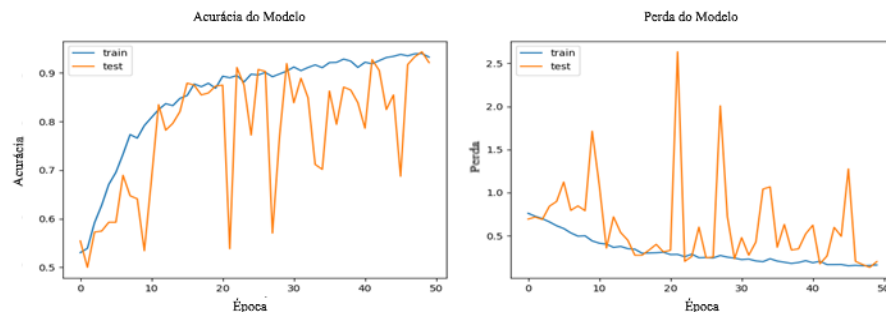


Figura 15.3 – Acurácia e Perda Fold 3 - ResNet50V2 (Lúcio Silva, 2021).

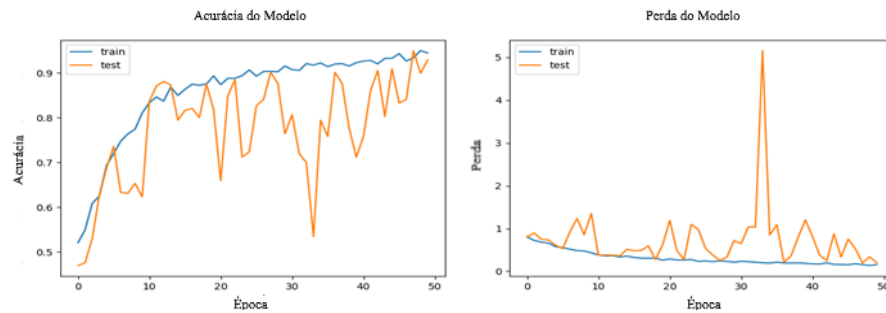


Figura 15.4 – Acurácia e Perda Fold 4 - ResNet50V2 (Lúcio Silva, 2021).

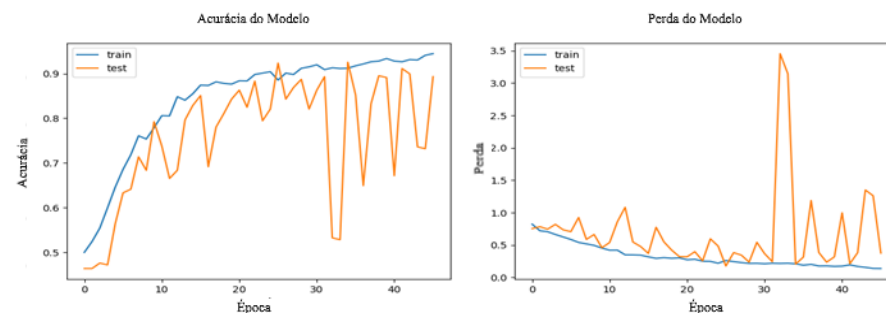


Figura 15.5 – Acurácia e Perda Fold 5 - ResNet50V2 (Lúcio Silva, 2021).

Os resultados alcançados pela InceptionV3 podem ser observados na Figura 16, que mostra sua matriz de confusão média, na qual podemos observar que a InceptionV3 estava incorreta em 30 casos, 13 casos deram falsos positivos para COVID-19 e 17 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da InceptionV3, mostrando a matriz de confusão e suas curvas de treinamento.

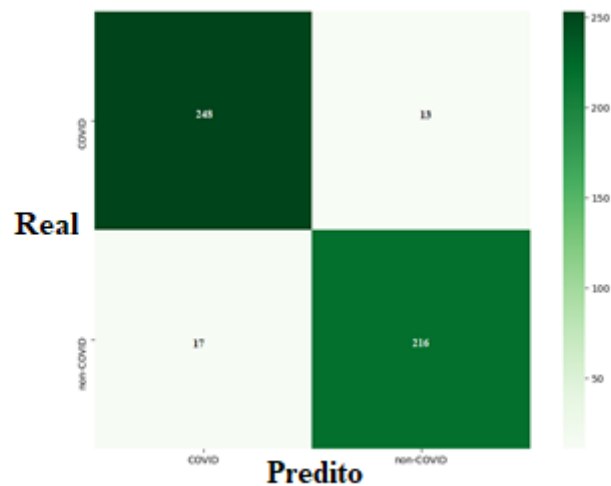


Figura 16 – Matriz de Confusão – InceptionV3 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 11, onde foram executadas com $k = 5$.

Tabela 11 – Folds InceptionV3.

Folds	Acurácia	Perda
Fold 1	0.951	0.138
Fold 2	0.945	0.139
Fold 3	0.931	0.162
Fold 4	0.957	0.140
Fold 5	0.911	0.229

Fonte: Lúcio Silva, 2021.

As Figuras 16.1 a 16.5 mostram a acurácia e a perda alcançadas pela InceptionV3. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda.

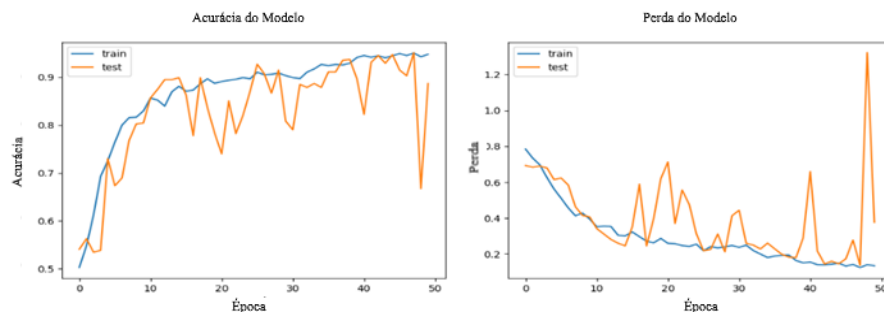


Figura 16.1 – Acurácia e Perda Fold 1 - InceptionV3 (Lúcio Silva, 2021).

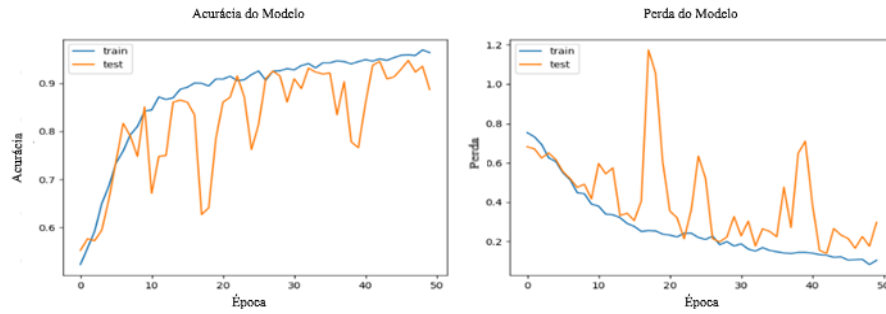


Figura 16.2 – Acurácia e Perda Fold 2 - InceptionV3 (Lúcio Silva, 2021).

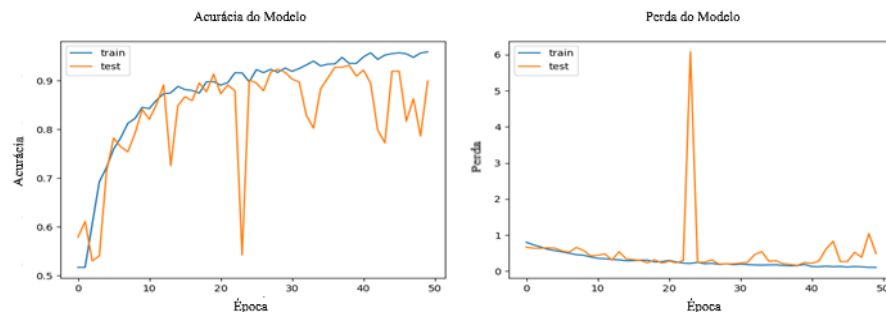


Figura 16.3 – Acurácia e Perda Fold 3 - InceptionV3 (Lúcio Silva, 2021).

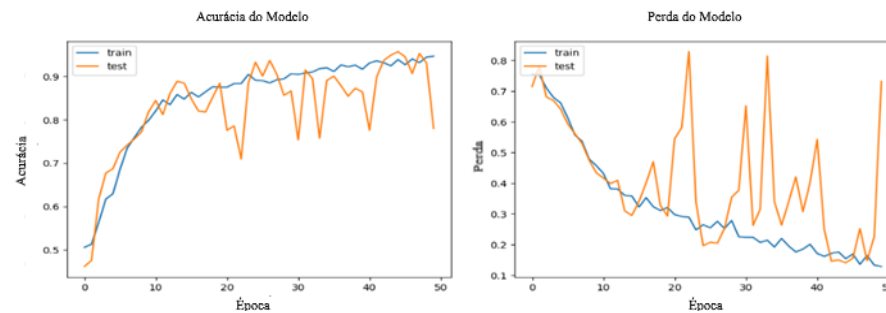


Figura 16.4 – Acurácia e Perda Fold 4 - InceptionV3 (Lúcio Silva, 2021).

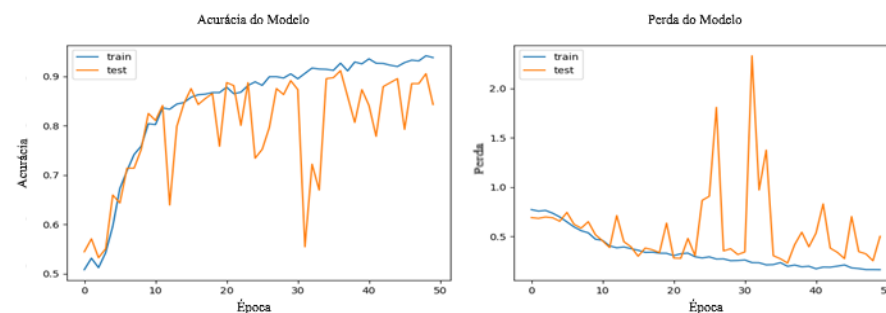


Figura 16.5 – Acurácia e Perda Fold 5 - InceptionV3 (Lúcio Silva, 2021).

Os resultados alcançados pela MobileNet podem ser observados na Figura 17, que mostra sua matriz de confusão média, na qual podemos observar que a MobileNet estava incorreta em 32 casos, 18 casos deram falsos positivos para COVID-19 e 14 casos de COVID-19 foram classificados erroneamente como não-COVID-19. Abaixo, detalhamos os resultados da MobileNet, mostrando a matriz de confusão e suas curvas de treinamento.

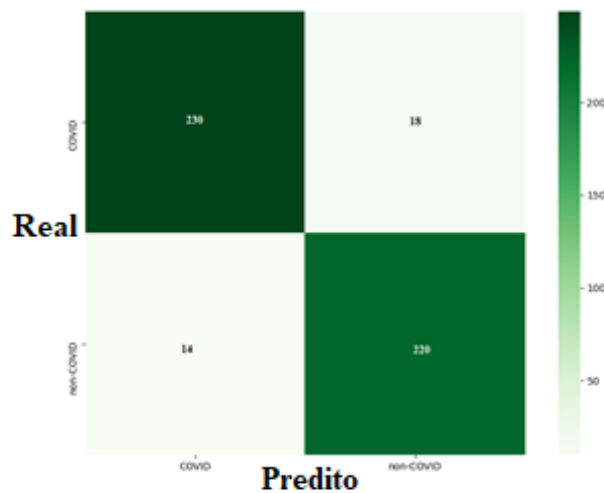


Figura 17 – Matriz de Confusão – MobileNet (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 12, onde foram executadas com $k = 5$.

Tabela 12 – Folds MobileNet.

Folds	Acurácia	Perda
Fold 1	0.949	0.156
Fold 2	0.955	0.112
Fold 3	0.925	0.183
Fold 4	0.937	0.152
Fold 5	0.905	0.223

Fonte: Lúcio Silva, 2021.

As Figuras 17.1 a 17.5 mostram a acurácia e a perda alcançadas pela MobileNet. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda. Em outras palavras, as curvas tendem a convergir.

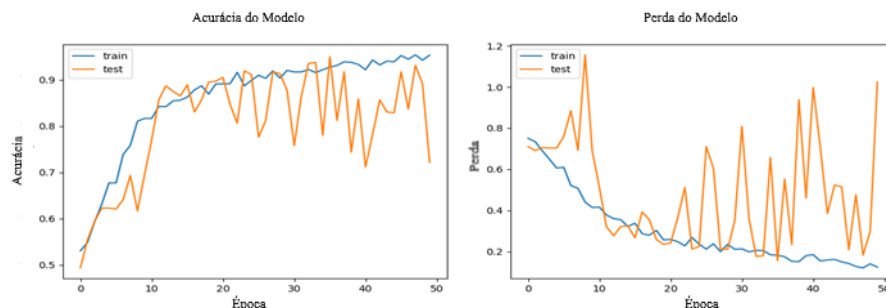


Figura 17.1 – Acurácia e Perda Fold 1 – MobileNet (Lúcio Silva, 2021).

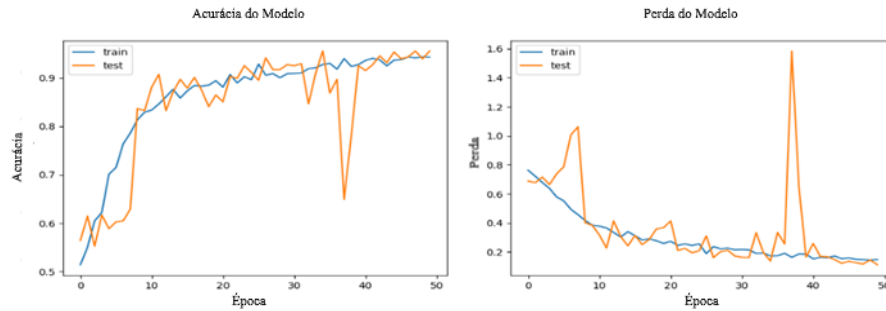


Figura 17.2 – Acurácia e Perda Fold 2 – MobileNet (Lúcio Silva, 2021).

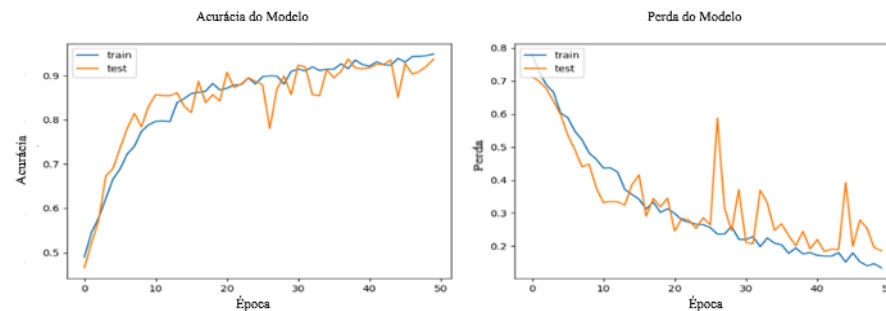


Figura 17.3 – Acurácia e Perda Fold 3 – MobileNet (Lúcio Silva, 2021).

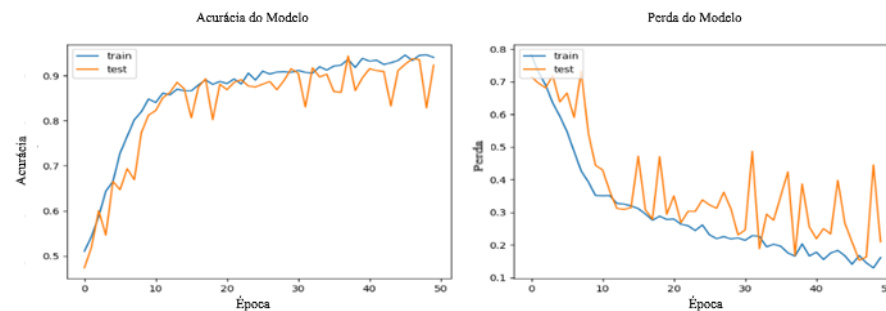


Figura 17.4 – Acurácia e Perda Fold 4 – MobileNet (Lúcio Silva, 2021).

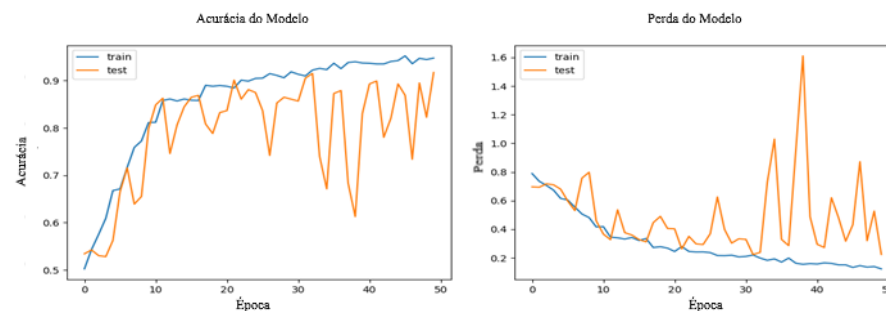


Figura 17.5 – Acurácia e Perda Fold 5 – MobileNet (Lúcio Silva, 2021).

A MobileNetV2 alcançou os piores resultados em relação as outras redes, como pode ser observado na Figura 18, que mostra sua matriz de confusão média, na qual podemos observar que a MobileNetV2 estava incorreta em 190 casos, 86 casos deram falsos positivos para COVID-19 e 104 casos de COVID-19 foram classificados erroneamente como não-

COVID-19. Abaixo, detalhamos os resultados da MobileNetV2, mostrando a matriz de confusão e suas curvas de treinamento.

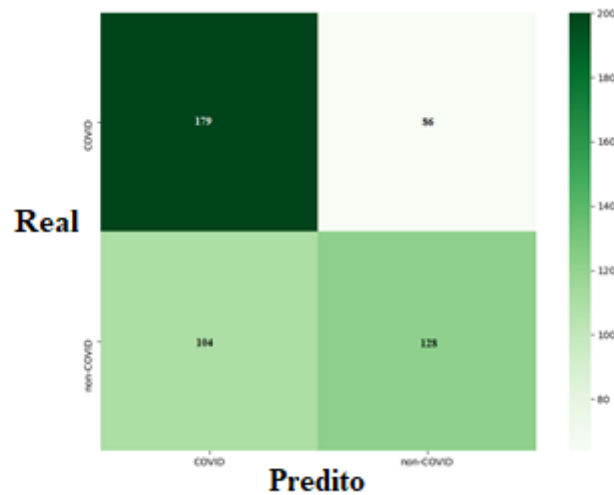


Figura 18 – Matriz de Confusão – MobileNetV2 (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 13, onde foram executadas com $k = 5$.

Tabela 13 – Folds MobileNetV2.

Folds	Acurácia	Perda
Fold 1	0.606	0.663
Fold 2	0.649	0.619
Fold 3	0.752	0.576
Fold 4	0.590	0.673
Fold 5	0.489	0.716

Fonte: Lúcio Silva, 2021.

As Figuras 18.1 a 18.5 mostram a acurácia e a perda alcançadas pela MobileNetV2. Como pode ser visto, as duas curvas não demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda. Em outras palavras, as curvas não tendem a convergir.

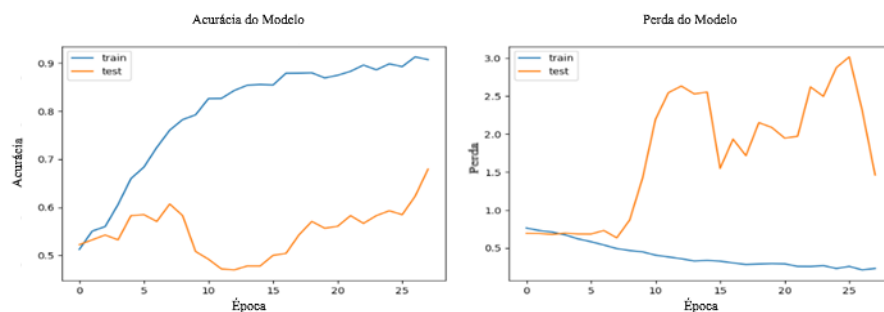


Figura 18.1 – Acurácia e Perda Fold 1 - MobileNetV2 (Lúcio Silva, 2021).

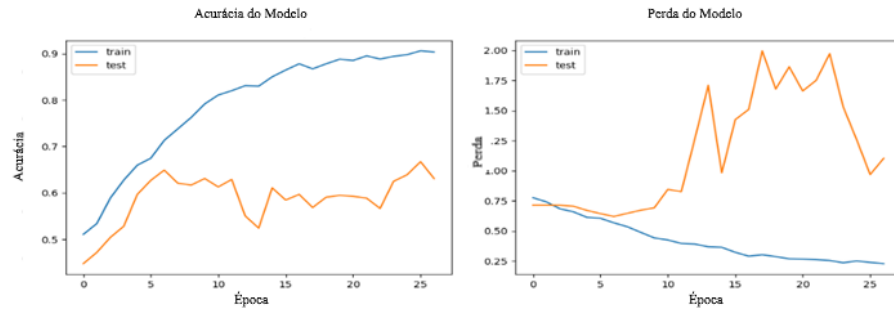


Figura 18.2 – Acurácia e Perda Fold 2 - MobileNetV2 (Lúcio Silva, 2021).

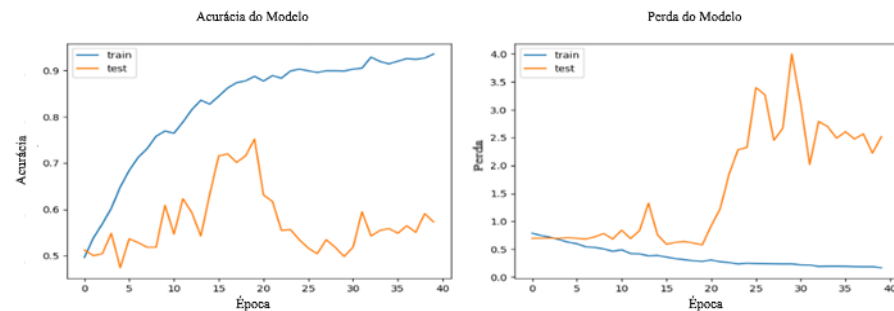


Figura 18.3 – Acurácia e Perda Fold 3 - MobileNetV2 (Lúcio Silva, 2021).

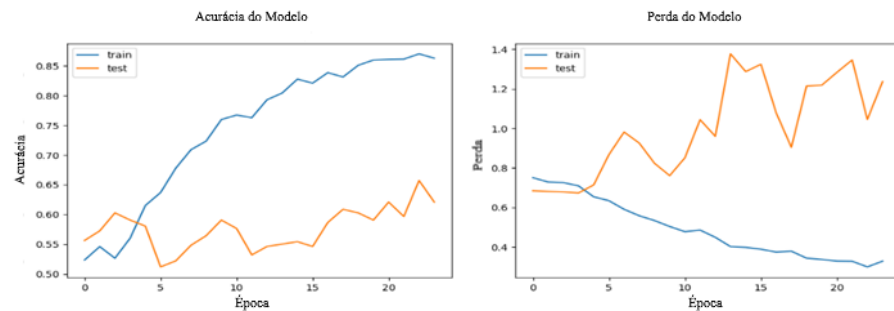


Figura 18.4 – Acurácia e Perda Fold 4 - MobileNetV2 (Lúcio Silva, 2021).

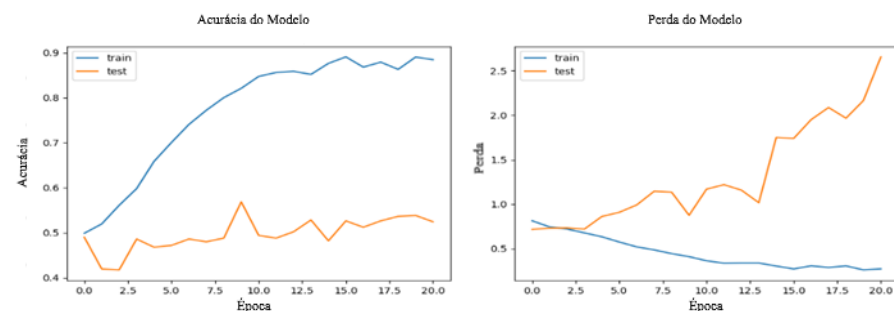


Figura 18.5 – Acurácia e Perda Fold 5 - MobileNetV2 (Lúcio Silva, 2021).

A EnsembleDVX apresentou o melhor resultado, a Figura 19 mostra sua matriz de confusão média, na qual podemos observar que a EnsembleDVX estava incorreta em apenas 11 casos, no entanto, 7 casos deram falsos positivos para COVID-19. Por outro lado, o fato notável é que apenas 4 casos de COVID-19 foram erroneamente classificados como não-COVID-19. Abaixo, detalhamos os resultados da EnsembleDVX, mostrando a matriz de confusão e suas curvas de treinamento.

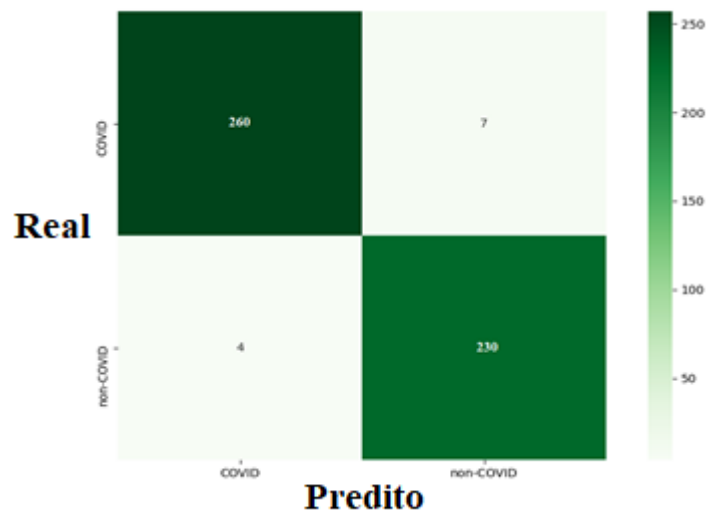


Figura 19 – Matriz de Confusão – EnsembleDVX (Lúcio Silva, 2021).

Os resultados da validação cruzada podem ser observados na Tabela 14, onde foram executadas com $k = 5$.

Tabela 14 – Folds EnsembleDVX.

Folds	Acurácia	Perda
Fold 1	0.980	0.074
Fold 2	0.976	0.085
Fold 3	0.978	0.076
Fold 4	0.978	0.079
Fold 5	0.976	0.071

Fonte: Lúcio Silva, 2021.

As Figuras 19.1 a 19.5 mostram a acurácia e a perda. Como pode ser visto, as duas curvas demonstram o comportamento esperado, que é atingir o valor um (100%) na acurácia e diminuir a zero (0%) a perda. Em outras palavras, as curvas tendem a convergir.

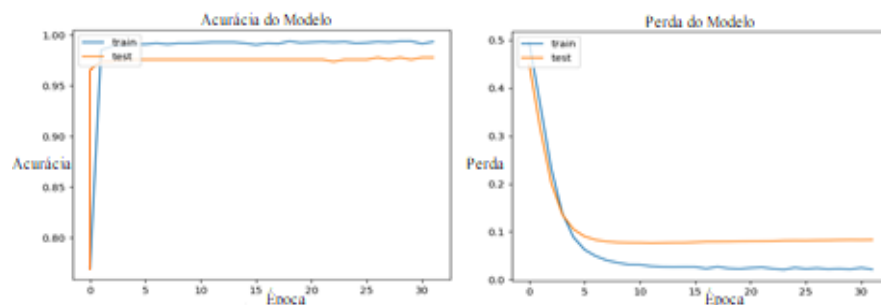


Figura 19.1 – Acurácia e Perda Fold 1 – EnsembleDVX (Lúcio Silva, 2021).

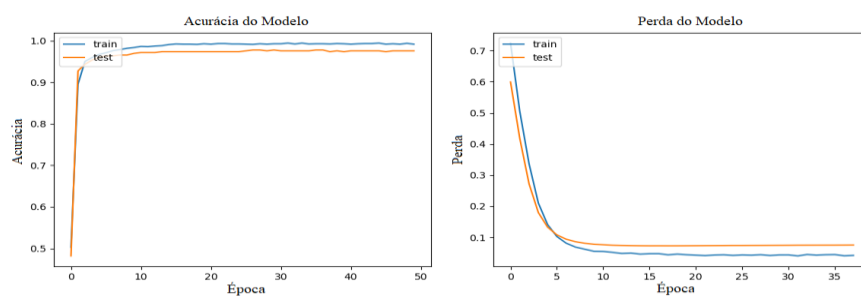


Figura 19.2 – Acurácia e Perda Fold 2 – EnsembleDVX (Lúcio Silva, 2021).

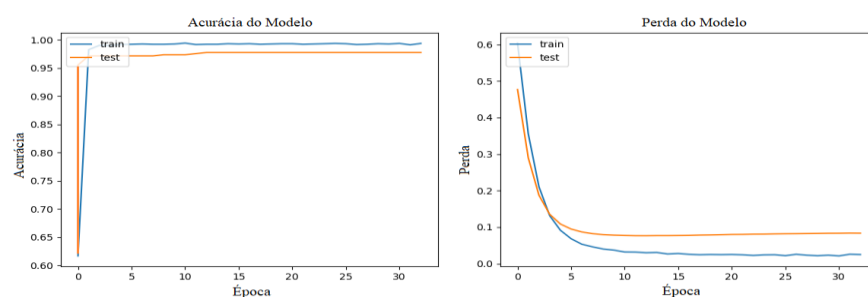


Figura 19.3 – Acurácia e Perda Fold 3 – EnsembleDVX (Lúcio Silva, 2021).

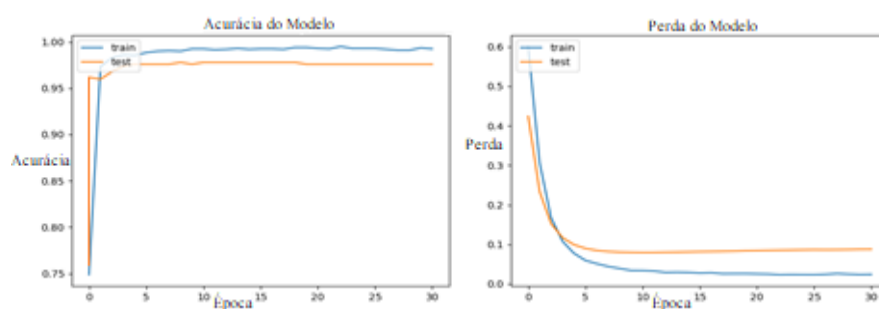


Figura 19.4 – Acurácia e Perda Fold 4 – EnsembleDVX (Lúcio Silva, 2021).

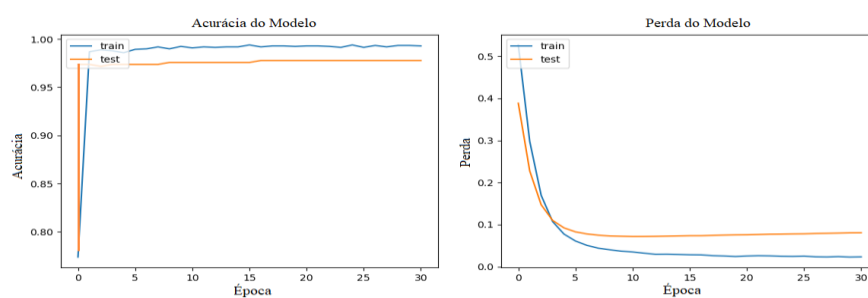


Figura 19.5 – Acurácia e Perda Fold 5 – EnsembleDVX (Lúcio Silva, 2021).

Como justificativa para a utilização de 3 modelos diferentes, também foi testado o agrupamento de dois em dois e os resultados são detalhados na tabela 14.1.

Tabela 14.1 – Agrupamento dois em dois EnsembleDVX.

Modelo	Média Acurácia	Média Precisão	Média Recall	Média F1_Score	Média Perda
EnsembleDVX	0.977\pm0.002	0.977\pm0.002	0.978\pm0.002	0.977\pm0.002	0.077 \pm 0.003
DenseNet169 + VGG16	0.972 \pm 0.001	0.972 \pm 0.001	0.973 \pm 0.001	0.972 \pm 0.001	0.080 \pm 0.002
DenseNet169 + Xception	0.976 \pm 0.001	0.976 \pm 0.001	0.976 \pm 0.001	0.976 \pm 0.001	0.073\pm0.002
VGG16 + Xception	0.975 \pm 0.002	0.974 \pm 0.002	0.952 \pm 0.002	0.974 \pm 0.002	0.093 \pm 0.002

Fonte: Lúcio Silva, 2021.

Como pode ser observado na Tabela 14.1, a EnsembleDVX alcançou resultados superiores quando comparado com as outras três combinações possíveis das três arquiteturas escolhidas, sendo inferior apenas na média da perda da DenseNet169 + Xception com uma pequena diferença de 0.004.

5 Conclusão

Os Sistemas de Diagnóstico de Doenças Auxiliados por Computador (CAD) auxiliam na classificação de imagens que pode determinar a presença ou ausência de uma doença de interesse. Os sistemas CAD se enquadram em um campo da Medicina 4.0 em que uma das aplicações tem o objetivo de desenvolver sistemas de apoio à decisão clínica. Estes sistemas de apoio à decisão estão relacionados aos procedimentos para melhorar as decisões médicas. Sendo assim, a detecção precoce de uma doença pode fornecer tempo suficiente para um tratamento ou ação bem sucedida.

Em geral, os algoritmos de Aprendizagem de Máquina (AM) são utilizados de forma a gerar classificadores para um conjunto de exemplos. Neste sentido, as técnicas de AM são empregadas na indução de um classificador que a partir de exemplos seja capaz de prever a classe de novos dados do montante em que foi treinado.

Um dos braços da Aprendizagem de Máquina são as abordagens de Aprendizagem Profunda, que são uma forma mais sofisticada e eficiente de lidar com grandes quantidades de dados. Contudo, as redes neurais profundas são computacionalmente mais complexas para o treinamento completo da rede. Como solução para esse problema uma técnica denominada de aprendizagem por transferência pode ser aplicada para minimizar esse problema. Na aprendizagem por transferência, a ideia é usar conhecimento prévio para resolver problemas semelhantes, ou seja, a aprendizagem por transferência é um método de reutilização de um modelo ou conhecimento para outra tarefa.

Além da aprendizagem por transferência, existe uma abordagem denominada de aprendizagem por agrupamento que se baseia na ideia de combinar diversos modelos de predição, treiná-los para uma mesma tarefa, e usá-los para produzir um modelo agrupado mais complexo, que é a soma de suas partes. Esta abordagem pode resultar em um melhor desempenho preditivo do que qualquer modelo único.

Esta pesquisa apresentou o desempenho de onze CNNs que foram treinadas usando aprendizagem por transferência e apresentou uma CNN com aprendizagem por agrupamento, denominada de EnsembleDVX, na classificação de imagens tomográficas, particularmente de pacientes diagnosticados com a COVID-19 e de pacientes que tiveram diagnóstico negativo para a COVID-19.

Os resultados mostraram que a EnsembleDVX atingiu melhores resultados em todas as médias das métricas utilizadas, além de ter alcançado o melhor resultado por fold. A EnsembleDVX classificou a maioria dos casos de pacientes com a COVID-19, errando apenas

quatro casos. Além disso, as curvas de acurácia e perda apresentaram um comportamento esperado indo em direção a uma acurácia com 100% e em direção a uma perda com 0%. Os resultados mostraram que a EnsembleDVX atingiu uma acurácia média de 97,7%, uma precisão média de 97,7%, um recall médio de 97,8% e F1 score médio de 97,7%.

O trabalho de pesquisa em questão foi aceito como capítulo de livro denominado como "*Using Convolutional Neural Networks for Classifying COVID-19 in Computerized Tomography Scans*" e será incluído no livro "*Recent Trends on Advancements of Image Processing and Its Applications*" correspondente a série "*EAI/Springer Innovations in Communications and Computing*". Além disso, será submetido um artigo estendido detalhando todos os resultados alcançados.

5.1 Trabalhos futuros

Nos trabalhos futuros serão testados as doze CNNs apresentadas em outros conjuntos de dados de imagens, como exemplo, em imagens radiográficas. Além disso, outras patologias serão incluídas no teste, com o objetivo de maximizar informações e resultados sobre o uso de redes neurais profundas e as abordagens de aprendizagem por transferência e aprendizagem por agrupamento na classificação de imagens médicas.

Além disso, poderá ser realizado o desenvolvimento de uma API com suporte as CNNs que apresentaram melhor desempenho com o objetivo de fornecer um caminho para que pesquisadores possam realizar testes em seus conjuntos de dados e que também possa ser consumida por outros aplicativos.

Referências

- ANGELOV, Plamen; ALMEIDA SOARES, Eduardo. SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification. medRxiv, 2020. Disponível em: <<https://covid-19.conacyt.mx/jspui/handle/1000/4168>>. Acesso em: 15 set. 2020.
- AMARO JÚNIOR, Edson; YAMASHITA, Helio. Aspectos básicos de tomografia computadorizada e ressonância magnética. Brazilian Journal of Psychiatry, v. 23, p. 2-3, 2001.
- ABADI, Martín et al. Tensorflow: A system for large-scale machine learning. In: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16). 2016. p. 265-283.
- BEYSOLOW II, T. (2017). Introduction to Deep Learning Using R: a step-by-step guide to learning and implementing Deep Learning Models Using R. Apress.
- BROWN, Gavin. Ensemble Learning. Encyclopedia of machine learning, v. 312, p. 15-19, 2010. Disponível em: <<http://www.cs.man.ac.uk/~gbrown/research/brown10ensemblelearning.pdf>>. Acesso em: 05 out. 2020.
- CARVALHO, A. C. P. L. F. Notas de aula: SCE5809 - Redes Neurais, Grupo de Inteligência Computacional, Pós-graduação em Computação, ICMC-USP, São Carlos-SP, 1999.
- COHEN J. P., MORRISON P., DAO L., "COVID-19 image data collection", arXiv:2003.11597, 2020. Disponível em: <<https://github.com/ieee8023/covid-chestxray-dataset>>. Acesso em: 10 de set. de 2020.
- CHETLUR, Sharan et al. cudnn: Efficient primitives for deep learning. arXiv preprint arXiv:1410.0759, 2014.
- Data Science Academy. Deep Learning Book, 2019. Disponível em: <<http://www.deeplearningbook.com.br/>>. Acesso em: 10 nov. 2020.
- DE WIT E, et al. SARS and MERS: recent insights into emerging coronaviruses. Nature Reviews Microbiology, 2016; 14, 523–534. Disponível em: <<https://doi.org/10.1038/nrmicro.2016.81>>. Acesso em: 15 nov. 2020.
- ESTEVES R. S., LORENA A. C., NASCIMENTO M. Z. Aplicação de técnicas de Aprendizado de Máquina na Classificação de Imagens Mamográficas, 2009. Disponível em: <http://ic.ufabc.edu.br/II_SIC_UFABC/resumos/paper_5_150.pdf> Acesso em: 15 set. 2020.
- FARO, S. H., & MOHAMED, F. B. (2010). BOLD fMRI - A Guide to Functional Imaging. New York: Springer Science.
- FORMISANO, E., SALLE, F., & GOEBEL, R. (2005). Fundamentals of Data Analysis Methods in Functional MRI. In L. Landini, V. Positano, & M. F. Santarelli, Advanced Image Processing in Magnetic Resonance Imaging (pp. 481-500). Taylor & Francis Group, LLC.
- KERAS. About. 2020. Disponível em: <https://keras.io/>, Acesso em: 10 set. 2020.

ISMAEL, Aras M.; ŞENGÜR, Abdulkadir. Deep learning approaches for COVID-19 detection based on chest X-ray images. *Expert Systems with Applications*, v. 164, p. 114054, 2021. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S0957417420308198>>. Acesso em: 26 fev. 2021.

LANGLEBEN, D. D.; MORIARTY, J. C. Using brain imaging for lie detection: Where science, law, and policy collide. *Psychology, Public Policy, and Law* 19, v. 2, n. 222. 2013. Disponível em: <https://pt.wikipedia.org/wiki/Imagem_por_ress%C3%B4nancia_magn%C3%A9tica_funcional#cite_note-1>. Acesso em: 15 set. 2020.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. *nature*, v. 521, n. 7553, p. 436-444, 2015.

LI, Lin et al. Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. *Radiology*, 2020. Disponível em: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233473/>>. Acesso em: 26 fev. 2020.

LIBRALÃO, G. L., OSHIRO, R. M., NETTO, A. V., CARVALHO, A. P. L. F., OLIVEIRA, M. C. F, 2003. Técnicas de Aprendizado de Máquina para análise de imagens oftalmológicas. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wim/2003/003.pdf>> Acesso em: 15 set. 2020.

NARIN, Ali; KAYA, Ceren; PAMUK, Ziyet. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *arXiv preprint arXiv:2003.10849*, 2020. Disponível em: <<https://arxiv.org/abs/2003.10849>>. Acesso em: 26 fev. 2021.

MELO, Fabrício Silva. Extração de relações a partir de dados não estruturados baseada em deep learning e supervisão distante. Dissertação (Mestrado) – Curso de Ciência da Computação, Universidade Federal de Sergipe, São Cristóvão, 2018.

M.E.H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z.B. Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N. Al-Emadi, M.B.I. Reaz, M. T. Islam, “Can AI help in screening Viral and COVID-19 pneumonia?” *IEEE Access*, Vol. 8, 2020, pp. 132665 - 132676. Disponível em: <<https://covid-19.conacyt.mx/jspui/handle/1000/4168>>. Acesso em: 26 fev. 2021.

McCULLOCH, W. S., and PITTS, W., “A logical calculus of the ideas immanent in nervous activity,” *Bulletin of Mathematical Biophysics*, vol. 5, pp. 115-133, 1943, Reprinted in [Anderson and Rosenfeld, 1988].

MINSKY, Marvin; PAPERT, Seymour A. *Perceptrons: An introduction to computational geometry*. MIT press, 2017.

MITCHELL, T. *Machine Learning*. McGraw Hill, 1997.

MONARD, M. & BARANAUSKAS, J. (2003). Conceitos sobre aprendizado de máquinas. Em *Sistemas Inteligentes: Fundamentos e Aplicações*, Cap. 4, Editora Manole, 512 Pp.

PAULING, L., & CORYELL, C. D. (1936). The magnetic properties and structure of hemoglobine, oxyhemoglobine and carbonmonoxyhemoglobine. *Proceedings of the National Academy of Sciences* Vol. 22, 210-216.

PEREZ, L. G. et al. Training an artificial neural network to discriminate between magnetizing inrush and internal faults, *IEEE Transactions on Power Delivery*, vol. 9, n. 1, pp. 434-441, 1994.

PHELPS, M. E., *PET Molecular Imaging and Its Biological Applications*. SPRINGER-VERLAG, NEW YORK, 2004. Disponível em: <<http://www.springer.com/la/book/9780387403595>>. Acesso em: 15 set. 2020.

POLIKAR, Robi. Ensemble learning. In: *Ensemble machine learning*. Springer, Boston, MA, 2012. p. 1-34. Disponível em: <https://link.springer.com/chapter/10.1007/978-1-4419-9326-7_1>. Acesso em: 21 fev. 2021.

PRAMPERO, P. S. (1998). Combinação de classificadores para reconhecimento de padrões, Dissertação (Mestrado), Instituto de Ciências Matemáticas e Computação - USP, São Carlos - SP, p 84., 12 out. 2017.

QUEIRÓS, G. C. de P. *Análise Computacional de Imagens de Ressonância Magnética Funcional*, 2011. Disponível em: <https://web.fe.up.pt/~tavares/downloads/publications/relatorios/Monografia_Gabriela_Queiros.pdf>. Acesso em: 15 set. 2020.

RAUBER, T. W. *Redes Neurais Artificiais*, 2005. Disponível em: <http://www.riopomba.ifsudestemg.edu.br/dcc/dcc/materiais/1926024727_reconhecimento-de-caracter2.pdf> Acesso em: 10 set. 2020.

ROSENBLATT, F., “The perceptron: A probabilistic model for information storage and organization in the brain,” *Psychological Review*, vol. 65, pp. 386-408, 1958.

SARKAR, D., BALI, R., and GHOSH, T. (2018). *Transfer Learning with Python*. Pack Publishing.

SCHNURR, H.P., ARONSKY, D., and WENKE, D. (2018). *MEDICINE 4.0—Interplay of Intelligent Systems and Medical Experts*, 51–63. Springer International Publishing, Cham.

SCHWARTZ, William Robson; PEDRINI, Hélio. Método para classificação de imagens baseada em matrizes de coocorrência utilizando características de textura. *III Colóquio Brasileiro de Ciências Geodésicas*, CuritibaPR, Brasil, p. 1, 2003. Disponível em: <https://homepages.dcc.ufmg.br/~william/papers/paper_2003_CBCG.pdf>. Acesso em: 05 out. 2020.

SILVA, Fillipe J.; JULIA, Rita M. S.; BARCELOS, Ayres. Diagnóstico de Imagens de Ressonância Magnética quanto a presença de Esclerose Múltipla por meio de Redes Neurais Artificiais. *BDBComp*, Minas Gerais, v. 1, n. 1. 2012. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/enia/2012/008.pdf>>. Acesso em: 15 set. 2020.

S.I. S. o. M. a. I. Radiology. (2020). *COVID-19 Database*. Disponível em: <<https://www.sirm.org/category/senza-categoria/covid-19/>>. Acesso em: 10 set. 2020.

SOARES, Eduardo et al. SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification. medRxiv, 2020.

TATIBANA, C. Y; KAETSU, D. Y. Uma introdução às redes neurais, 2000. Disponível em: <<http://www.din.uem.br/ia/neurais>>. Acesso em: 10 set. 2020.

TINOIS, Eduardo. Imagem Funcional PET E fMRI. Tecnologia para a Saúde V, Campinas, v. 1, n. 1, outubro. 2005. Disponível em: <http://www.multiciencia.unicamp.br/artigos_05/a_02_05.pdf>. Acesso em: 15 set. 2020.

VAID, Shashank; KALANTAR, Reza; BHANDARI, Mohit. Deep learning COVID-19 detection bias: accuracy through artificial intelligence. International Orthopaedics, v. 44, p. 1539-1542, 2020. Disponível em: <<https://link.springer.com/content/pdf/10.1007/s00264-020-04609-7.pdf>>. Acesso em: 26 fev. 2021.

WIDROW, B., and HOFF, M. E., “Adaptive switching circuits,” in 1960 WESCON Convention Record, New York, 1960.

WOLF, B. and SCHOLZE, C. (2017). “medicine 4.0”. Current Directions in Biomedical Engineering, 3.

WOLPERT, David H. Stacked generalization. Neural networks, v. 5, n. 2, p. 241-259, 1992. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S0893608005800231>>. Acesso em: 05 out. 2021.

World Health Organization (2020). Situation Report - 158. <https://www.who.int/publications/m/item/weekly-operational-update-on-covid-19---22-february-2021>, Acesso em: 22 fev. 2021.

XIN, Yang et al. Aprendizado de máquina e métodos de aprendizado profundo para segurança cibernética. Acesso IEEE , v. 6, p. 35365-35381, 2018.

ZHENG, Chuansheng et al. Deep learning-based detection for COVID-19 from chest CT using weak label. MedRxiv, 2020. Disponível em: <<https://www.medrxiv.org/content/medrxiv/early/2020/03/26/2020.03.12.20027185.full.pdf>>. Acesso em: 26 fev. 2021.