

Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Engenharia Eletrônica

**Comparação de Modelos de Aprendizado de Máquina para  
Classificação de Plantas quanto ao seu Mecanismo de  
Resistência à Presença de Alumínio no Solo**

**Autora: Alessandra Maia Freire  
Orientador: Dr. Renan Utida Barbosa Ferreira**

**Brasília, DF  
2022**





Alessandra Maia Freire

**Comparação de Modelos de Aprendizado de Máquina  
para Classificação de Plantas quanto ao seu Mecanismo  
de Resistência à Presença de Alumínio no Solo**

Monografia submetida ao curso de graduação em (Engenharia Eletrônica) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia Eletrônica).

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Dr. Renan Utida Barbosa Ferreira

Coorientador: Dr. Luciano Paulino da Silva

Brasília, DF

2022

---

Alessandra Maia Freire

Comparação de Modelos de Aprendizado de Máquina para Classificação de Plantas quanto ao seu Mecanismo de Resistência à Presença de Alumínio no Solo/  
Alessandra Maia Freire. – Brasília, DF, 2022-

111 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. Renan Utida Barbosa Ferreira

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA , 2022.

1. Redes Neurais Convolucionais. 2. Mecanismo de Resistência ao Alumínio em Plantas. I. Dr. Renan Utida Barbosa Ferreira. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Comparação de Modelos de Aprendizado de Máquina para Classificação de Plantas quanto ao seu Mecanismo de Resistência à Presença de Alumínio no Solo

CDU 02:141:005.6

---

Alessandra Maia Freire

# **Comparação de Modelos de Aprendizado de Máquina para Classificação de Plantas quanto ao seu Mecanismo de Resistência à Presença de Alumínio no Solo**

Monografia submetida ao curso de graduação em (Engenharia Eletrônica) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia Eletrônica).

Trabalho aprovado. Brasília, DF, 6 de maio de 2022:

---

**Dr. Renan Utida Barbosa Ferreira**  
Orientador

---

**Dr. Cristiano Jacques Miosso**  
Convidado 1

---

**Dr. Bruno Luigi Macchiavello  
Espinoza**  
Convidado 2

Brasília, DF  
2022



# Agradecimentos

Agradeço aos meus orientadores, Luciano Paulino da Silva e Renan Utida Barbosa Ferreira, por me incentivarem, me ajudarem, me apoiarem e me guiarem durante todo o processo desse trabalho.

Agradeço à equipe do Laboratório de Nanobiotecnologia da Embrapa, à qual fui apresentada durante o processo de realização desse projeto, que me ajudou bastante no entendimento da multidisciplinaridade da pesquisa, sempre com empolgação e carinho, me incentivando a inovar e buscar ultrapassar fronteiras do conhecimento.

Agradeço à minha família que esteve torcendo por mim, me aconselhando, me apoiando e me assistindo em todos os momentos da vida desde sempre.

Não posso deixar de agradecer também aos meus amigos, que estiveram me acompanhando nesse processo, fazendo parte de grandes etapas da minha vida.





*“O aumento do conhecimento  
é como uma esfera dilatando-se no espaço:  
quanto maior a nossa compreensão,  
maior o nosso contato com o desconhecido.  
(Blaise Pascal)*



# Resumo

A disponibilidade de alumínio solúvel em solos ácidos é um grande inibidor do crescimento das plantas, sendo um dos principais limitadores da produtividade agrícola nesses solos. Em contrapartida, na flora do Cerrado brasileiro são encontradas diversas espécies de plantas resistentes à presença desse metal no solo. Os mecanismos de resistência ao alumínio são divididos em duas categorias: a das plantas excludentes, que impedem a entrada do metal nas suas partes aéreas, e a das plantas acumuladoras, que incorporam o metal dentro de si. Esse projeto visa realizar uma comparação entre determinadas arquiteturas de aprendizado profundo de máquina para classificação de imagens microscópicas de folhas de plantas específicas, quanto à presença ou ausência de alumínio nas folhas das plantas. Para isso, foi utilizado um banco de imagens privado, desenvolvido pelo Laboratório de Nanobiotecnologia da Embrapa Recursos Genéticos e Biotecnologia, de aproximadamente 1040 imagens de folhas de plantas do Cerrado resistentes à presença de alumínio no solo, caracterizadas como acumuladoras ou excludentes.

Para os treinos, foram utilizadas as arquiteturas Resnet-RS-152, Resnet-RS-101 e Resnet-RS-50, todas testadas com modelos com e sem o uso de parâmetros pré-treinados. Todas as partes de programação, treinamento e testes das arquiteturas foram realizadas no ambiente computacional Kaggle, utilizando o Pytorch como principal biblioteca de implementação desse processo. Foram realizadas duas classificações distintas no mesmo banco de imagens: a classificação binária das plantas das imagens em acumuladoras ou excludentes de alumínio e a classificação em famílias de plantas, com 12 famílias no total. A implementação das redes foi feita e validada utilizando a função de perda de entropia cruzada e por métricas de desempenho *k-fold*, com 10 *folds*.

Nos resultados, puderam ser observadas as diferenças de acurácias obtidas entre os modelos com parâmetros pré-treinados e os modelos sem esse pré-treino para ambos conjuntos de classificações. Para a classificação binária, o modelo sem pré-treino ResNet-RS-152 obteve uma acurácia total de 89,13% e desvio padrão de  $\pm 1,14\%$  enquanto o modelo de ResNet-RS-152 pré-treinado alcançou 98,37% de acurácia, com desvio padrão de  $\pm 0,62\%$ . Na classificação das imagens em famílias de plantas, o modelo sem pré-treino que alcançou melhor resultado foi o da ResNet-RS-101, com 88,65% de acurácia e desvio padrão de  $\pm 1,04\%$  enquanto o modelo pré-treinado com melhores resultados foi o da ResNet-RS-50, com 98,17% de acurácia total e desvio padrão de  $\pm 0,97\%$ .

Neste trabalho também foram avaliados os resultados das métricas de desempenho precisão, sensibilidade e *f1-score* para cada classe de ambos conjuntos, com o objetivo de analisar mais profundamente a qualidade de classificação dos modelos e identificar possíveis dificuldades ou irregularidades nas classificações. Ao final, são gerados gráficos jun-

tamente com as tabelas de resultados para uma melhor visualização desses resultados e são avaliadas futuros incrementos e implementações para o projeto.

Os resultados mostraram que, para a máquina, é possível classificar as imagens microscópicas de plantas em agrupamentos de acordo com suas características de resistência ao alumínio, como acumuladoras e excludentes. É mostrada também a possibilidade de classificação das imagens do banco de acordo com as famílias às quais as plantas pertencem, mesmo havendo mais de uma espécie por família no banco de dados. Como para este trabalho haviam apenas duas famílias com espécies acumuladoras e dez famílias com espécies excludentes, não se sabe se as características encontradas pela máquina são referentes à resistência ao alumínio ou se são características que as famílias das plantas do banco de dados possuem em comum. Com a expansão do banco de dados, acrescentando mais espécies de plantas de ambas classificações, será possível tirar mais conclusões quanto à esse fato. Mesmo assim, o trabalho apresentado pode auxiliar no desenvolvimento de uma grande diversidade de pesquisas onde a característica de acumulação ou exclusão de metal em plantas é relevante. Tais quais podem incluir desde a avaliação de plantas comestíveis que, por acumularem determinados metais podem causar problemas de saúde ao ser humano (BOJÓRQUEZ-QUINTAL et al., 2017; GUERRA et al., 2012) até pesquisas relacionadas à fitomineração, que é a extração de metais pesados do solo por determinados grupos de plantas acumuladoras, favorecendo também a remediação desses solos (fitorremediação) (AKINBILE; MAKHUBELA; AMBUSHE, 2021; ASHRAF et al., 2019; SHEORAN; SHEORAN; POONIA, 2009).

**Palavras-chaves:** Classificação de Imagens. Alumínio. Plantas. Redes Neurais. Aprendizado de Máquina. Resnet-RS.

# Abstract

Soluble aluminum availability in acid soils is a major inhibitor of plant growth, being one of the main agricultural productivity limiting factor in these type of soils. In the brazilian Cerrado flora, several species of plants have developed resistance mechanisms to Al under those conditions. These resistance mechanisms are divided into two categories: plants that are able to prevent the aluminum to entry in its aerial parts have the excluding mechanism, while those that are able to incorporate Al within them are called accumulators. This project aims to make a comparison between certain deep machine learning architectures for classifying microscopic images of specific plant leaves, for the presence or absence of aluminum in the leaves of the plants. For this purpose, it was used an image database developed in the Nanobiotechnology Laboratory of Embrapa Genetic Resources and Technology. This database contains approximately 1040 images of leaves of cerrado plants which are resistant to the presence of alluminum in the soil, classified as as alluminum accumulators or excluders.

For training, the ResnetRS-152, ResnetRS-101 and ResnetRS-50 architectures were used, all tested with and without pre-trained parameters. The programming, training and testing steps of the models implementation were carried out in the Kaggle computing environment, using Pytorch as the main implementation library for this process. The image database was duplicated and organized to be used in two groups of classifications: a binary classification, in which the plant from the image will be classified as an aluminum accumulator or excluder and a multiclass classification, in which the images are classified according to the plants families, with 12 classes in total. The implementation of the networks was done and validated using the cross entropy loss function and the k-fold performance metrics, with 10 folds.

In the results, it was possible to observe the difference between the accuracies obtained from the models with pre-trained parameters and the ones without pre-training for both classification groups. For the binary classification, the model without pre-training, ResNet-Rs-152, obtained a total accuracy of 89,13% and a standard deviation of  $\pm 1.14\%$  while the same model with pre-trained parameters reached 98,37% accuracy, with a standard deviation of  $\pm 0,62\%$ . For the multiclass classification into plant families, the model without pre-training that achieved the highest result was the ResNet-RS-101, with 88,65% of accuracy and standard deviation of  $\pm 1.04\%$  while the pre-trained model with the best results was the ResNet-RS-50, with 98,17% total accuracy and a standard deviation of  $\pm 0,97\%$ .

In this work, the results of performance metrics precision, sensitivity and f1-score for each class of both groups are also evaluated, with the objective of analyzing more deeply the classification quality of the models and identifying possible difficulties or irregularities in

the classifications. Finally, graphs are used along with the tables with the resulting values for a better visualization of these values. At the end, future increments and implementations for the project are evaluated.

The results showed that, for a machine learning program, it is possible to classify the microscopic images of plants according to their aluminum resistance features, such as accumulators and excluders. The possibility of image classification according to the families to which the plants belong is also shown, even for cases in the database where there is more than one species per family. Because there were only two families with accumulator species and ten families with excluder species in the database, it is not known if the features found by the machine are related to aluminum resistance or if they are characteristics that the families of plants in the database have in common. With the expansion of the database, adding more species of plants of both classifications, it would be possible to draw more conclusions regarding this fact. Even so, the presented work could aid the development of a great diversity of researches where the feature of metal accumulation or exclusion in plants is relevant. Such range of researches could include from the evaluation of edible plants that, by accumulating certain metals, can cause health problems to humans (BOJÓRQUEZ-QUINTAL *et al.*, 2017; GUERRA *et al.*, 2012) to researches related to phytomining, which is the extraction of heavy metals from the soil by certain groups of accumulator plants, also favoring the remediation of these soils (phytoremediation) (AKINBILE; MAKHUBELA; AMBUSHE, 2021; ASHRAF *et al.*, 2019; SHEORAN; SHEORAN; POONIA, 2009).

**Key-words:** Image Classification. Aluminum. Plants. Neural Networks. Machine Learning. Resnet-RS.

# Lista de ilustrações

Figura 1 – Representação de uma imagem digital bidimensional, onde cada quadrado da malha representa um pixel da imagem. . . . .	34
Figura 2 – Representação de uma operação de convolução de uma imagem 4x4 com um filtro de tamanho 2x2 . . . . .	34
Figura 3 – Representação do modelo de um <i>perceptron</i> . . . . .	36
Figura 4 – Elementos básicos de uma CNN. . . . .	37
Figura 5 – operação de subamostragem máximo utilizando uma janela 2x2 . . . .	38
Figura 6 – Algumas das funções de ativação mais comuns que são utilizadas em redes neurais profundas. . . . .	39
Figura 7 – Representação das conexões dos neurônios de uma camada totalmente conectada de uma CNN após o procedimento de achatamento. . . . .	40
Figura 8 – Representação de aplicação de <i>Dropout</i> . À direita: um modelo de rede com duas camadas ocultas. À esquerda: um exemplo de rede com <i>Dropout</i> aplicado (SRIVASTAVA et al., 2014). . . . .	41
Figura 9 – Representação das etapas de treino de uma rede neural, adaptado de (YAMASHITA; NISHIO; DO, 2018). . . . .	42
Figura 10 – Exemplo de curva de perda apresentando sobreajuste: A curva de perda dos dados de validação está muito acima e divergente da curva de perda dos dados de treino, que continua a decrescer. Na seção 2.4.2 é visto um pouco mais sobre esse tipo de curva. . . . .	43
Figura 11 – Ilustração da aplicação do método <i>k-fold</i> com $k=5$ . . . . .	43
Figura 12 – Representação da configuração da arquitetura VGGNet modelo VGG16 (KHAN et al., 2018) . . . . .	45
Figura 13 – Exemplo de um bloco residual ResNet. . . . .	46
Figura 14 – Representação da arquitetura da ResNet-50, onde os blocos em azul contém as convoluções dos blocos residuais e CTC representa as camadas totalmente conectadas. . . . .	47
Figura 15 – Comparação de diferentes métodos de escalonamento de redes neurais, adaptado de (TAN; LE, 2020). . . . .	47
Figura 16 – Representação do bloco residual invertido, onde H,W e F representam as dimensões dos dados em questão, H representa a altura, W a largura e F a profundidade. . . . .	48
Figura 17 – Ilustração de uma arquitetura ResNet-RS . . . . .	49
Figura 18 – Representação de uma matriz de confusão com os valores focados na classe "0", organizado em Positivo Verdadeiro, Positivo Falso, Negativo Falso e Falso Verdadeiro. . . . .	50

Figura 19 – Representação de uma matriz de confusão para a classificação das classes A, B e C, onde (A), (B) e (C) possuem os valores Positivo Verdadeiro, Positivo Falso, Negativo Falso e Falso Verdadeiro focados em suas respectivas classes. . . . .	51
Figura 20 – Exemplo de curvas de perda de validação e treino . . . . .	52
Figura 21 – Exemplo de curva de aprendizado de acurácia . . . . .	52
Figura 22 – Imagens retiradas do banco de dados. Cada figura representa uma das espécies de plantas do banco de dados, essas espécies são agrupadas em famílias e as famílias, agrupadas de acordo com o mecanismo de resistência ao alumínio. . . . .	53
Figura 23 – (A) - Imagem de resolução original 1936x1936 pixels (reescalada). (B) - Mesma Imagem de (A), porém com 224x224 pixels de resolução e escala original. . . . .	57
Figura 24 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-50. . . . .	62
Figura 25 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-50. . . . .	62
Figura 26 – Matriz de confusão gerada a partir dos dados de teste do modelo ResNet-RS-50 para todos os <i>folders</i> . . . . .	63
Figura 27 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-101. . . . .	64
Figura 28 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-101. . . . .	64
Figura 29 – Matriz de confusão gerada a partir dos dados de teste da ResNet-RS-101 para todos os <i>folders</i> . . . . .	65
Figura 30 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-152. . . . .	66
Figura 31 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-152. . . . .	67
Figura 32 – Matriz de confusão gerada a partir dos dados de teste do modelo ResNet-RS-152 para todos os <i>folders</i> . . . . .	67
Figura 33 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-50 com parâmetros pré-treinados. . . . .	69
Figura 34 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-50. . . . .	69



Figura 35 – Matriz de confusão gerada a partir resultados dos testes do modelo pré-treinado ResNet-RS-50, para todos os <i>folds</i> . . . . .	70
Figura 36 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-101 com parâmetros pré-treinados. . . . .	71
Figura 37 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-101. . . . .	71
Figura 38 – Matriz de confusão gerada a partir dos dados gerais de teste do modelo pré-treinado ResNet-101 para todos os <i>folds</i> . . . . .	72
Figura 39 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-152 com parâmetros pré-treinados. . . . .	72
Figura 40 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-152. . . . .	73
Figura 41 – Matriz de confusão gerada a partir dos dados gerais de teste do modelo pré-treinado ResNet-RS-152 para todos os <i>folds</i> . . . . .	73
Figura 42 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-50 com as classes de famílias de plantas. . . . .	75
Figura 43 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas em todos os <i>folds</i> para o modelo ResNet-RS-50 sem pré-treino. . . . .	75
Figura 44 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-101 para as classes de famílias de plantas. . . . .	77
Figura 45 – Matriz de Confusão com os dados de teste do modelo ResNet-RS-101 para a classificação em famílias de plantas para todos os <i>folds</i> . . . . .	77
Figura 46 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-152 para as classes de famílias de plantas. . . . .	79
Figura 47 – Matriz de Confusão com os dados de teste para a classificação entre as famílias das plantas de todos os <i>folds</i> para o modelo ResNet-RS-152. . . . .	79
Figura 48 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-50 pré-treinado para as classes de famílias de plantas. . . . .	81
Figura 49 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas de todos os <i>folds</i> para o modelo ResNet-RS-50 pré-treinado. . . . .	81

Figura 50 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-101 pré-treinado para as classes de famílias de plantas. . . . .	83
Figura 51 – Matriz de Confusão com os dados de teste para a classificação entre as famílias das plantas do banco de dados de todos os <i>folds</i> para o modelo ResNet-RS-101 sem pré-treinado. . . . .	83
Figura 52 – Gráfico com as curvas de perda dos processos de validação (laranja) e treino (azul) para o modelo ResNet-RS-152 pré-treinado, para as classes de famílias de plantas. . . . .	85
Figura 53 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas de todos os <i>folds</i> para o modelo ResNet-RS-152 pré-treinado. . . . .	85
Figura 54 – Tradução dos dados da Tabela 31, comparando as acurácias obtidas com tempo que cada modelo demorou para terminar de rodar. As barras coloridas representam os desvios padrão de cada modelo ResNet-RS. . . . .	88
Figura 55 – Comparação dos valores de precisão, em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32. . . . .	90
Figura 56 – Comparação dos valores de sensibilidade, em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32. . . . .	90
Figura 57 – Comparação dos valores de <i>f1-score</i> , em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32. . . . .	91
Figura 58 – Tradução dos dados da Tabela 33 comparando as acurácias obtidas com tempo que cada modelo demorou para terminar de rodar. As barras coloridas representam os desvios padrão de cada modelo ResNet-RS. . . . .	92
Figura 59 – Valores de precisão obtidos para em cada modelo ResNet-RS para cada família de planta. . . . .	109
Figura 60 – Valores de sensibilidade obtidos para em cada modelo ResNet-RS para cada família de planta. . . . .	110
Figura 61 – Valores de <i>f1-score</i> obtidos para em cada modelo ResNet-RS para cada família de planta. . . . .	111

# Lista de tabelas

Tabela 1	– Tabela da relação da quantidade de imagens, de famílias e de espécies para cada mecanismo de resistência. . . . .	54
Tabela 2	– Relação entre família e mecanismo de resistência correspondente para cada espécie de planta, onde A = Acumuladora e E = Excludente, (OLIVEIRA et al., 2019) cujas imagens das folhas estão sendo utilizadas no banco. . . . .	54
Tabela 3	– Proporção da quantidade de imagens do banco de dados para cada família de planta, com o número de imagens por lote sendo representado por uma média da quantidade de imagens por lote de cada família de acordo com a proporção. . . . .	55
Tabela 4	– Comparação entre os modelos de ResNet-RS, e os modelos das EfficientNets nos quesitos de Parâmetros, FLOPs, Latência de TPU, Latência de GPU, e Precisão atingida no ImageNet sem o uso de dados extras de treino, adaptado de (BELLO et al., 2021) . . . . .	56
Tabela 5	– Comparação entre as plataformas Kaggle e Google Colab/Google Colab Pro. . . . .	58
Tabela 6	– Comparação entre as bibliotecas mais utilizadas para aprendizado profundo de máquina. . . . .	59
Tabela 7	– Resultados gerais de acurácia a ResNet-RS-50, com seus respectivos desvios padrão em relação aos valores obtidos em todos os <i>folders</i> . . . . .	63
Tabela 8	– Resultados do modelo ResNet-RS-50, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os <i>folders</i> . . . . .	63
Tabela 9	– Resultados gerais de acurácia a ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores obtidos em todos os <i>folders</i> . . . . .	65
Tabela 10	– Resultados do modelo ResNet-RS-101, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os <i>folders</i> . . . . .	65
Tabela 11	– Resultados de acurácia da ResNet-RS-152, com seus respectivos desvios padrão em relação aos valores obtidos em todos os <i>folders</i> . . . . .	67
Tabela 12	– Resultados do modelo ResNet-RS-152, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os <i>folders</i> . . . . .	68
Tabela 13	– Resultados de acurácia da ResNet-RS-50 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os <i>folders</i> . . . . .	68

Tabela 14 – Resultados específicos do modelo ResNet-RS-50 pré-treinado, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação às métricas obtidas em todos os <i>folds</i> . . . . .	69
Tabela 15 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-101 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os <i>folds</i> . . . . .	70
Tabela 16 – Resultados do modelo ResNet-RS-101 pré-treinado, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os <i>folds</i> . . . . .	71
Tabela 17 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-152 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os <i>folds</i> . . . . .	74
Tabela 18 – Resultados do modelo ResNet-RS-152 pré-treinado, com as métricas de desempenho por classe e respectivos desvios padrão em relação a todos os <i>folds</i> . . . . .	74
Tabela 19 – Resultados de acurácia para a ResNet-RS-50, com seus respectivos desvios padrão em relação às acurácias de todos os <i>folds</i> . . . . .	76
Tabela 20 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-50 e seus respectivos desvios padrão. . . . .	76
Tabela 21 – Resultados das métricas de desempenho para a ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores de todos os <i>folds</i> 78	78
Tabela 22 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-101 e seus respectivos desvios padrão. . . . .	78
Tabela 23 – Resultados de acurácia para a ResNet-RS-152, com seus respectivos desvios padrão em relação aos resultados de todos os <i>folds</i> . . . . .	80
Tabela 24 – Métricas de desempenho por classe do modelo ResNet-RS-152 e seus respectivos desvios padrão. . . . .	80
Tabela 25 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-50 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os <i>folds</i> . . . . .	82
Tabela 26 – Tabela com as métricas de desempenho por classe do modelo pré-treinado ResNet-RS-50 e seus respectivos desvios padrão. . . . .	82
Tabela 27 – Resultados de acurácia do modelo pré-treinado ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores obtidos em todos os <i>folds</i> . . . . .	84
Tabela 28 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-101 pré-treinado e seus respectivos desvios padrão. . . . .	84

Tabela 29 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-152 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os <i>folds</i> . . . . .	86
Tabela 30 – Tabela com as métricas de desempenho por classe do modelo pré-treinado ResNet-RS-152 e seus respectivos desvios padrão. . . . .	86
Tabela 31 – Dados resultantes de acurácias gerais, desvios padrão e tempo total para o treino, validação e testes para os 10 <i>folds</i> de todos os modelos ResNet-RS utilizados para a classificação das imagens das plantas em acumuladoras ou excludentes. . . . .	88
Tabela 32 – Dados resultantes das métricas de desempenho por classe e desvios padrão para todos os modelos ResNet-RS utilizados para a classificação das imagens das plantas em acumuladoras ou excludentes. Valores obtidos como médias de seus respectivos valores de todos os <i>folds</i> . . . . .	89
Tabela 33 – Dados resultantes de acurácias gerais, desvios padrão e tempo total para o treino, validação e testes para os 10 <i>folds</i> de todos os modelos ResNet-RS utilizados na classificação das imagens de acordo com as famílias das plantas. . . . .	92
Tabela 34 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Apocynaceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Valores obtidos como médias de todos os <i>folds</i> . A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie <i>Hancornia speciosa</i> , da família Apocynaceae. . . . .	103
Tabela 35 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Caryocaraceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os <i>folds</i> . A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie <i>Caryocar brasiliense</i> , da família Caryocaraceae. . . . .	103
Tabela 36 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Celastraceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os <i>folds</i> . A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie <i>Salacia crassifolia</i> , da família Celastraceae. . . . .	104
Tabela 37 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Clusiaceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os <i>folds</i> . A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie <i>Kielmeyera coriacea</i> , da família Clusiaceae. . . . .	104

- Tabela 38 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Fabaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Hymenaea stigonocarpa*, da família *Fabaceae*. 105
- Tabela 39 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Lythraceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Lafoensia pacari*, da família *Lythraceae*. . 105
- Tabela 40 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Malvaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Eriotheca gracilipes*, da família *Malvaceae*. 106
- Tabela 41 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Melastomataceae* com imagens de plantas acumuladoras para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Pleroma stenocarpum*, da família *Melastomataceae*. . . . . 106
- Tabela 42 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Myrtaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Psidium myrsinoides*, da família *Myrtaceae*. 107
- Tabela 43 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Ochnaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Ouratea hexasperma*, da família *Ochnaceae*. 107
- Tabela 44 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Styracaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Styrax ferrugineus*, da família *Styracaceae*. 108

Tabela 45 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Vochysiaceae* com imagens de plantas acumuladoras para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Qualea parviflora*, da família *Vochysiaceae*. 108





# Lista de abreviaturas e siglas

Al	Alumínio
CNN	Redes Neurais Convolucionais, do inglês <i>Convolutional Neural Network</i>
ILSVRC	Desafio de Reconhecimento Visual em Grande Escala da ImageNet, do inglês <i>ImageNet Large Scale Visual Recognition Challenge</i>
RGB	Vermelho, Verde e Azul, do inglês <i>Red, Green, Blue</i>
sRGB	Padrão Vermelho, Verde e Azul, do inglês <i>standard Red, Green and Blue</i>
RNA	Redes Neurais Artificiais
MSE	Erro Médio Quadrático, do inglês <i>Mean Squared Error</i>
CE	Entropia Cruzada, do inglês <i>Cross-Entropy loss</i>
CCE	Entropia Cruzada Categórica, do inglês <i>Categorical Cross-Entropy loss</i>
BCE	Entropia Cruzada Binária, do inglês <i>Binary Cross-Entropy loss</i>
PMC	Perceptron de Múltiplas Camadas
VGG	Grupo de Geometria Visual, do inglês <i>Visual Geometry Group</i>
ResNet	Rede Residual, do inglês <i>Residual Network</i>
FLOPs	Operações de ponto flutuante, do inglês <i>Floating-point Operations</i>
ReLu	Unidade Linear Retificada, do inglês <i>Rectified Linear Unit</i>
SE	Função de compressão e agitação, do inglês <i>Squeeze-and-Excitation function</i>
TPU	Unidade de Processamento de Tensor, do inglês <i>Tensor Processing Unit</i>
GPU	Unidade de Processamento Gráfico, do inglês <i>Graphics Processing Unit</i>
CPU	Unidade Central de Processamento, do inglês <i>Central Processing Unit</i>



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>29</b>
<b>1.1</b>	<b>Contextualização</b>	<b>29</b>
<b>1.2</b>	<b>Problemas e Proposta de Pesquisa</b>	<b>30</b>
<b>1.3</b>	<b>Objetivos</b>	<b>31</b>
1.3.1	Objetivo Geral	31
1.3.2	Objetivos Específicos	31
<b>1.4</b>	<b>Estrutura da Dissertação</b>	<b>32</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>33</b>
<b>2.1</b>	<b>Alumínio nas Plantas e Mecanismos de Resistência</b>	<b>33</b>
<b>2.2</b>	<b>Processamento de imagens digitais</b>	<b>33</b>
2.2.1	Convolução de imagens digitais	34
<b>2.3</b>	<b>Aprendizado de Máquina</b>	<b>35</b>
2.3.1	Classificação automática de imagens digitais	35
2.3.2	Redes Neurais Artificiais	35
2.3.2.1	Redes Neurais Convolucionais – CNN	37
2.3.2.1.1	Camada Convolucional	37
2.3.2.1.2	Camada de Subamostragem	38
2.3.2.1.3	Não-linearidade	39
2.3.2.1.4	Camada Totalmente Conectada	40
2.3.2.1.5	<i>Dropout</i>	40
2.3.2.1.6	Normalização de Lote	41
2.3.2.2	Treinamento	41
2.3.2.2.1	Sobreajuste	42
2.3.2.2.2	<i>K-Fold</i>	42
2.3.2.2.3	Função de Perda	43
2.3.2.2.4	Regularização	44
2.3.2.2.5	<i>RandAugment</i>	45
2.3.2.3	Arquiteturas de CNNs	45
2.3.2.3.1	VGGNet	45
2.3.2.3.2	ResNet	46
2.3.2.3.3	EfficientNet	47
2.3.2.3.4	ResNet-RS	48
<b>2.4</b>	<b>Ferramentas de Validação</b>	<b>49</b>
2.4.1	Métricas de Desempenho	49

2.4.2	Curvas de Aprendizado . . . . .	52
<b>3</b>	<b>METODOLOGIA . . . . .</b>	<b>53</b>
<b>3.1</b>	<b>Banco de Dados . . . . .</b>	<b>53</b>
<b>3.2</b>	<b>Arquiteturas Utilizadas . . . . .</b>	<b>55</b>
3.2.1	ResNet-RS . . . . .	55
<b>3.3</b>	<b>Pré-processamento . . . . .</b>	<b>56</b>
<b>3.4</b>	<b>Recursos Computacionais . . . . .</b>	<b>57</b>
3.4.1	<i>Hardware</i> . . . . .	57
3.4.2	<i>Software</i> . . . . .	58
<b>3.5</b>	<b>Aplicação e Métricas de desempenho . . . . .</b>	<b>60</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÕES . . . . .</b>	<b>61</b>
<b>4.1</b>	<b>Acumuladoras e Excludentes . . . . .</b>	<b>61</b>
4.1.1	Sem pré-treino . . . . .	61
4.1.1.1	ResNet-RS-50 . . . . .	61
4.1.1.2	ResNet-RS-101 . . . . .	63
4.1.1.3	ResNet-RS-152 . . . . .	66
4.1.2	Pré-treinadas . . . . .	68
4.1.2.1	ResNet-RS-50 . . . . .	68
4.1.2.2	ResNet-RS-101 . . . . .	70
4.1.2.3	ResNet-RS-152 . . . . .	72
<b>4.2</b>	<b>Famílias de Plantas . . . . .</b>	<b>74</b>
4.2.1	Sem pré-treino . . . . .	74
4.2.1.1	ResNet-RS-50 . . . . .	74
4.2.1.2	ResNet-RS-101 . . . . .	77
4.2.1.3	ResNet-RS-152 . . . . .	78
4.2.2	Pré-treinadas . . . . .	80
4.2.2.1	ResNet-RS-50 . . . . .	81
4.2.2.2	ResNet-RS-101 . . . . .	83
4.2.2.3	ResNet-RS-152 . . . . .	85
<b>4.3</b>	<b>Resultados Gerais e Discussões . . . . .</b>	<b>87</b>
4.3.1	Classificação em Acumuladoras e Excludentes . . . . .	87
4.3.2	Classificação em famílias de plantas . . . . .	91
<b>5</b>	<b>CONCLUSÃO . . . . .</b>	<b>95</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>97</b>

**ANEXOS**

**101**

**ANEXO A – RESULTADOS GERAIS: MÉTRICAS DE DESEMPENHO DAS CLASSIFICAÇÕES EM FAMÍLIAS . . . . 103**



# 1 Introdução

## 1.1 Contextualização

A toxicidade do alumínio é uma das principais limitações para o cultivo, em solos ácidos, de diversas espécies de importância agronômica (CONCEIÇÃO; SERENO; NETO, 2008). O alumínio (Al) é um dos metais mais abundantes da crosta terrestre, sendo encontrado principalmente complexado na forma de silicatos e óxidos de alumínio, que não apresentam riscos de toxicidade para as plantas (OLIVEIRA et al., 2019). Em solos ácidos porém, uma fração desse Al se torna solúvel na forma de cátions  $A^{+1}$ ,  $A^{+2}$  e  $A^{+3}$  potencialmente tóxicos para as plantas. Por esse fator, solos ácidos são praticamente sinônimos de toxicidade por alumínio. No caso do Brasil, dois terços do território é constituído por solos ácidos (VITORELLO; CAPALDI; STEFANUTO, 2005). O alumínio solúvel, mesmo em concentrações micromolares, inibe o crescimento da raiz como um todo das plantas sensíveis a este metal. A grande maioria das plantas de interesse agrônomico, quando cultivadas na presença de Al catiônico, apresentam raízes menores e encarquilhadas, deficientes na absorção de água e nutrientes e, conseqüentemente, menos produtivas e mais sujeitas às doenças (OLIVEIRA et al., 2019). Entretanto, seu mecanismo de fitotoxicidade ainda não é totalmente compreendido (PANDA; MATSUMOTO, 2007).

O alumínio (Al) é o terceiro metal mais abundante da crosta terrestre. Nos solos ácidos o Al encontra-se solúvel na forma de cátions que são tóxicos para a grande maioria das plantas cultivadas. Entretanto, as plantas nativas do Cerrado são resistentes ao alumínio catiônico presente nos solos ácidos desse Bioma. Essas plantas são classificadas em duas categorias: acumuladoras e excludentes. As plantas que possuem tolerância interna, chamadas acumuladoras, possuem mecanismos que envolvem o acúmulo de alumínio em sua parte aérea (tronco, folhas, flores e frutos) de modo que o metal não prejudique o desenvolvimento da planta (HARIDASAN, 2008). As plantas que possuem tolerância externa possuem mecanismos de exclusão do alumínio, impedindo que o metal atinja a parte aérea do vegetal. Contudo, ainda há um número limitado de estratégias para identificá-las e geralmente os métodos são complexos e trabalhosos. (OLIVEIRA et al., 2019).

A classificação automática de imagens de organismos biológicos vem sendo muito utilizada e melhorada nas mais diversas áreas, como medicina e agropecuária. Nos casos de classificação de organismos biológicos microscópicos, como as células, mesmo quando os classificadores não são capazes de refletir a diversidade biológica desse meio, eles ainda conseguem muitas vezes obter um desempenho melhor do que um especialista humano (OEI et al., 2019). O uso de redes neurais para classificações de imagens é uma prática

em constante expansão e crescimento. Um exemplo disso é o desafio ILSVRC - *ImageNet Large Scale Visual Recognition Challenge* ou Desafio de Reconhecimento Visual em Grande Escala da ImageNet, que é referência na classificação e detecção de objetos em centenas de categorias distintas e vem sendo executado até o presente (RUSSAKOVSKY et al., 2015). A eficiência e precisão das últimas gerações de modelos participantes vêm mostrando o grande progresso que tem sido feito na área de reconhecimento e classificação de imagens ao longo dos anos, possibilitando inúmeras aplicações (RUSSAKOVSKY et al., 2015).

## 1.2 Problemas e Proposta de Pesquisa

Atualmente, para se verificar se a planta possui ou não algum mecanismo de resistência à presença de Al em solo ácido, é necessário realizar a observação do desenvolvimento da planta, principalmente de suas raízes, nessas condições de solo (PANDA; MATSUMOTO, 2007). A classificação das plantas quanto ao método de resistência ao Al é normalmente possível a partir de testes em laboratório, onde é medida a quantidade de alumínio presente na matéria orgânica seca da planta ou por métodos histoquímicos com corantes específicos (COTTA et al., 2008). A determinação do mecanismo de resistência de uma planta à presença de alumínio no solo requer procedimentos que necessitam de tempo, recursos infraestruturais de laboratório e conhecimento especializado, sendo uma área de estudos de grande importância devido ao impacto do alumínio em plantações do mundo todo (OLIVEIRA et al., 2019). Mesmo dentro das classificações de plantas acumuladoras, seus mecanismos de resposta à presença do metal no solo podem variar bastante nos quesitos de funcionamento e estratégia. As concentrações de Al dentro da matéria orgânica também variam entre as espécies dentro dessa classificação (HARIDASAN, 2008; OLIVEIRA et al., 2019).

Para se gerar alternativas para a agropecuária em solos ácidos, faz-se necessário primeiramente compreender os mecanismos de resistência ao Al das plantas nativas de lugares com estas condições de solo para desenvolver plantas tolerantes ao metal. A análise de imagens de nível microscópico de folhas de plantas resistentes à presença de alumínio no solo focando na classificação quanto ao mecanismo de resistência dessa planta ao metal agilizaria o processo de classificação além de poder ser ajustado e expandido tanto em questão de banco de dados quanto em informações de saída da máquina após o treino, abrindo portas para novas possibilidades para pesquisas nessa área, não apenas para o Al, mas também para outros metais como níquel e ouro.

Este trabalho traz como proposta a análise do desempenho de classificação de imagens microscópicas de folhas de plantas resistentes à presença de alumínio no solo quanto ao seu mecanismo de resistência (acumuladoras ou excludentes) utilizando métodos de



aprendizado de máquina. Apesar de existirem inúmeras análises e classificações de plantas utilizando aprendizado de máquina com as mais variadas metodologias e arquiteturas (SUNNY, 2020; SUN et al., 2017), não foram encontradas pesquisas que fizessem classificações das plantas de acordo com seu mecanismo de resistência ao alumínio a partir de imagens microscópicas de folhas, sendo assim, foi necessário fazer correlações entre áreas e/ou tipos de classificações que se assemelhassem ao objetivo desta pesquisa para a escolha das arquiteturas de redes neurais.

## 1.3 Objetivos

### 1.3.1 Objetivo Geral

O objetivo desta pesquisa é analisar diferentes modelos de arquiteturas de aprendizado de máquina, visando diferenciar as imagens de plantas de acordo com sua estratégia de resistência ao alumínio no solo, classificando-as como acumuladoras ou excludentes. Para uma melhor compreensão geral, os resultados serão comparados com os de classificação das imagens em famílias de plantas.

### 1.3.2 Objetivos Específicos

Para que seja atingido o objetivo geral, serão abordados os seguintes objetivos intermediários:

- Organizar o banco de dados para se aplicar os dois tipos de classificação: a classificação binária em acumuladoras (A) e excludentes (E) e a classificação de múltiplas classes para as famílias de plantas.
- Implementar diferentes modelos de arquiteturas ResNet-RS nas mesmas condições de hardware e software para comparação de desempenho;
- Testar o funcionamento das arquiteturas com o banco de dados selecionado;
- Utilizar as métricas de validação k-fold para validação da implementação;
- Após o treinamento e testes, verificar e comparar os resultados de acurácia, precisão, sensibilidade e *f1-score*, juntamente com seus respectivos desvios padrão;
- Fazer uma análise das relações entre os resultados e desvios-padrão obtidos dos modelos pré-treinados e os sem pré-treino;
- Analisar as relações entre as métricas de desempenho obtidas e desvios-padrão dos modelos com classificação por família e dos modelos com classificação pela característica de resistência ao alumínio.

## 1.4 Estrutura da Dissertação

Este documento será organizado da seguinte forma nas próximas seções. No Capítulo 2, Referencial Teórico, são abordados alguns conceitos para o melhor entendimento do trabalho, tais como noções gerais do mecanismo de resistência das plantas ao alumínio solúvel no solo, processamento digital de imagens e redes neurais convolucionais com algumas técnicas de treinamento e ferramentas de validação.

No Capítulo 3 são apresentadas as metodologias, onde são mostrados os meios e recursos utilizados para a realização do trabalho, como a base de dados e as arquiteturas utilizadas, os recursos computacionais e métricas de desempenho.

No Capítulo 4 são apresentados os resultados e discussões para cada caso aplicado e os resultados e discussões dos dados gerais. Por fim, no Capítulo 5, são apresentadas as conclusões juntamente com as possíveis contribuições do estudo apresentado e sugestões de complementações futuras no trabalho.

## 2 Referencial Teórico

### 2.1 Alumínio nas Plantas e Mecanismos de Resistência

A presença de alumínio em solos ácidos é um fator que limita o crescimento de plantas que não são adaptadas a esse meio (ROUT; SAMANTARAY; DAS, 2001). Vários estudos vêm mostrando que a inibição do crescimento das raízes das plantas é o sintoma visível mais rápido de se observar da toxicidade do alumínio. Essa má-formação das raízes acaba resultando em uma deficiência hídrica, mineral e má fixação da planta ao solo (ECHART; CAVALLI-MOLINA, 2001).

A tolerância ao Al é geralmente analisada em nível celular (CONCEIÇÃO; SERENO; NETO, 2008). As plantas que resistem à presença do metal por meio de processos de exclusão não apresentam níveis significantes do metal em sua matéria orgânica, pois elas possuem mecanismos fisiológicos em suas raízes que impedem a absorção do metal pela planta (OLIVEIRA et al., 2019). As plantas que possuem tolerância interna, ou acumuladoras, possuem mecanismos cujo funcionamento resulta em acúmulo de alumínio na raiz, no caule e nas folhas do vegetal (HARIDASAN, 2008).

### 2.2 Processamento de imagens digitais

De maneira básica, processamento digital de uma imagem se refere à tradução de uma imagem para o meio digital (GONZALEZ; WOODS, 2009). No caso da escala de cinza, a imagem é convertida em pontos de uma malha discreta, com cada ponto de intensidade representado por um número finito adequado de valores ou, para o caso de imagens coloridas, por um vetor com componentes de valores adequados (JÄHNE, 1995). Uma imagem em escala de cinza pode ser definida como uma função  $f(x,y)$ , onde as variáveis  $x$  e  $y$  são coordenadas espaciais do plano e  $f$  é a intensidade do pixel, ou o nível de cinza. O pixel é definido como a menor unidade que compõe uma imagem digital. Quando as coordenadas e a intensidade representam quantidades finitas e discretas, chamamos esse conjunto de pixels de imagem digital (GONZALEZ; WOODS, 2009). Na Figura 1 é ilustrado um exemplo de imagem digital formada por pixels.

Para imagens coloridas, é bastante utilizado o sistema RGB - *Red*, *Green* e *Blue*, onde um pixel é representado por um vetor cujas componentes representam as intensidades luminosas das cores vermelho, verde e azul, respectivamente. Com isso, a formação das imagens coloridas se dá pela composição de três imagens monocromáticas, cada uma representando um extremo de intensidade em RGB (QUEIROZ; GOMES, 2001).

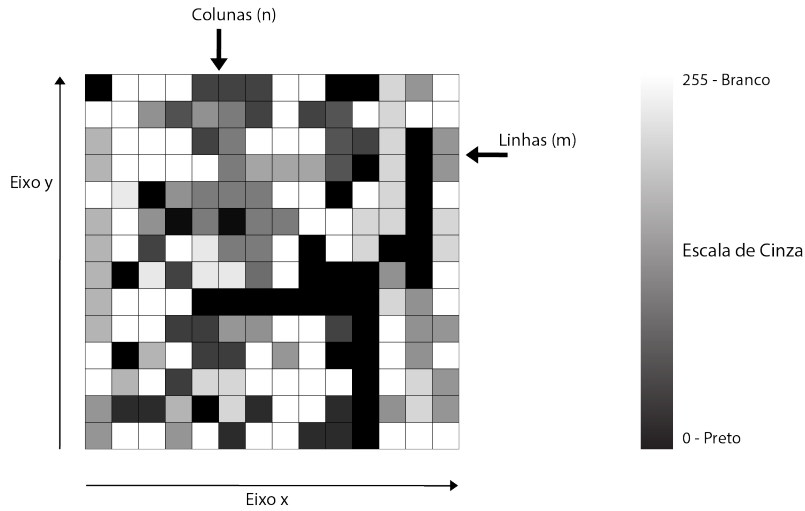


Figura 1 – Representação de uma imagem digital bidimensional, onde cada quadrado da malha representa um pixel da imagem.

### 2.2.1 Convolução de imagens digitais

A convolução discreta pode ser descrita como uma operação linear de somatória do produto entre duas funções ao longo de uma região que elas sobrepõem, em razão do deslocamento que existe entre elas. Dessa forma, uma bidimensional do filtro  $w(x,y)$ , com uma imagem de entrada  $f(x,y)$  pode ser representada pela Equação 2.1,

$$[\mathbf{w} * \mathbf{f}] = \sum_{s=-a}^a \sum_{t=-b}^b \mathbf{w}(\mathbf{s}, \mathbf{t})\mathbf{f}(\mathbf{x} - \mathbf{s}, \mathbf{y} - \mathbf{t}), \tag{2.1}$$

supondo que o filtro possui resposta finita ao impulso, centralizado na origem e de ordem par na horizontal e na vertical, com  $a$  e  $b$  representando as metades das ordens em cada direção.

Para a área de processamento de imagens, a convolução é utilizada para filtrar imagens com o objetivo de se extrair informações de interesse. Esse processo pode ser descrito como a aplicação de uma máscara sobre cada pixel e sua vizinhança, como representado na Figura 2, deslocando-se um pixel por vez.

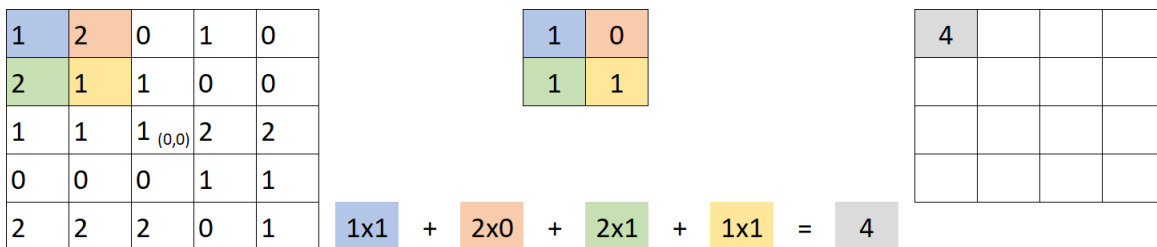


Figura 2 – Representação de uma operação de convolução de uma imagem 4x4 com um filtro de tamanho 2x2

## 2.3 Aprendizado de Máquina

### 2.3.1 Classificação automática de imagens digitais

O aprendizado de máquina como é conhecido hoje evoluiu do estudo de reconhecimento de padrões e da teoria do aprendizado computacional dentro do campo de inteligência artificial. O aprendizado de máquina explora a construção e o estudo de algoritmos que podem aprender e fazer previsões sobre determinados dados (KOHAVI; PROVOST, 1998). Estes algoritmos operam construindo um modelo a partir de entradas de treinamento para fazer previsões ou decisões baseadas nestes dados ao em vez de seguir instruções estáticas de programação (BISHOP, 2006).

As tarefas realizadas utilizando aprendizado de máquina podem ser divididas em três grandes categorias (RUSSEL; NORVIG, 2003):

- **Aprendizagem supervisionada:** são apresentados à máquina os exemplos de entradas e as saídas desejadas correspondentes, fornecidas por um supervisor da máquina, processo comparado com uma dinâmica entre professor e estudante, onde o objetivo é aprender uma regra geral que mapeia as entradas nas saídas. Um exemplo é o treino de um banco de imagens com etiquetas das classificações às quais elas pertencem com o objetivo de aprender os padrões entre as classificações das imagens.
- **Aprendizagem não-supervisionada:** nenhuma etiqueta é dada ao algoritmo de aprendizado, deixando-o sozinho para encontrar a estrutura em sua entrada. A aprendizagem não supervisionada pode ser um objetivo por si só, utilizado, por exemplo, para descobrir padrões ocultos em dados, ou um meio para um fim específico.
- **Aprendizagem reforçada:** Um programa de computador interage com um ambiente dinâmico e repetitivo, no qual ele deve realizar um determinado objetivo como, por exemplo, dirigir um veículo ou jogar um jogo, sem um “professor” para explicitar se a máquina se aproxima ou não de um determinado objetivo.

### 2.3.2 Redes Neurais Artificiais

Uma rede neural artificial - RNA, consiste em uma estrutura onde o processamento é distribuído por um grande número de unidades menores interligadas. Elas foram criadas tomando como inspiração as características de alto grau de paralelismo e conectividade dos sistemas biológicos, principalmente do sistema nervoso. No ponto de vista computacional, o neurônio é enxergado como um modelo de sistema que realiza o processamento de determinados dados sobre uma ou várias entradas com o objetivo de se gerar uma saída (FURTADO, 2019). O esquemático da Figura 3 representa o modelo de neurônio proposto

por McCulloch e Pitts em 1943, (MCCULLOCH; PITTS, 1943), chamado também de *perceptron*.

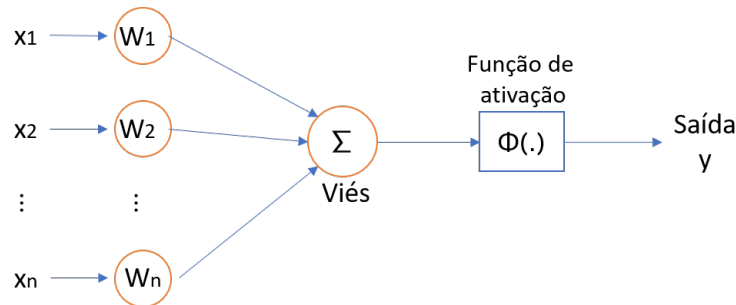


Figura 3 – Representação do modelo de um *perceptron*.

Uma rede neural pode apresentar diferentes resultados dependendo de sua arquitetura e necessidade de aplicação. As arquiteturas fundamentais de RNA podem ser divididas em três classes: redes neurais *feedforward* de camada única, redes neurais *feedforward* de múltiplas camadas e redes recorrentes ou realimentadas (FURTADO, 2019). A rede neural *feedforward* de camada única, também chamada de *perceptron*, mostrada na Figura 3, é o tipo mais simples de Rede Neural. Ela possui uma única camada de nós de saída, onde as entradas são alimentadas diretamente nas saídas por uma série de pesos e a soma dos produtos dos pesos e das entradas é calculada em cada nó. O valor então deve passar por uma função de ativação, que ativa ou não o neurônio na saída de acordo com seus parâmetros (GALLO, 2015). Nesse modelo, cada valor  $x_i$  do vetor de entrada representa um pixel da imagem que se deseja analisar. Os pesos, dados por  $\{W_1, W_2, \dots, W_n\}$ , são valores escalares que são multiplicados aos valores de entrada para torná-los menos relevantes ou destacá-los. O viés é um parâmetro adicional utilizado para aumentar o grau de liberdade da função e ajustar a saída junto à soma ponderada das entradas para o neurônio. A função de ativação, dada por  $\varphi(\cdot)$ , é utilizada para se decidir se o neurônio será ou não ativado. Por fim, a saída do *perceptron* é dada por um valor de saída  $y$ .

Uma rede neural *feedforward* de múltiplas camadas também possui como base o *perceptron*, porém inclui uma ou mais camadas, cada uma contendo um ou mais neurônios, podendo apresentar números distintos de neurônios entre as camadas. Nessa arquitetura, as camadas contidas entre a entrada e a saída da rede são chamadas de camadas ocultas (GALLO, 2015). Uma grande diferença entre as redes neurais recorrentes e as de *feedforward* é que as redes recorrentes permitem a formação de círculos de realimentação, podendo realizar uma realimentação do neurônio com sua própria saída, o que também é chamado de *self-feedback* (FURTADO, 2019).

### 2.3.2.1 Redes Neurais Convolucionais – CNN

Rede Neural Convolutional, ou CNN, do inglês *Convolutional Neural Network*, trata de um modelo de rede neural que pode receber como entrada uma imagem, ou determinados tipos de dados, atribuir importâncias a determinadas características desses dados e ser capaz de retornar na saída certas informações específicas, como classificações (INDOLIA et al., 2018). Embora esse modelo de rede tenha sido proposto por Y. Lecun em 1989 (LELUN, 1989), sua implementação exigia uma grande complexidade computacional, o que só foi alcançado posteriormente com o uso de GPUs -*graphics processing units* no lugar das CPUs *central processing unit* para realizar o processamento de dados da rede (OH; JUNG, 2004).

Em partes, uma CNN aprende a reconhecer determinadas características de baixo nível em suas camadas iniciais, seguidas por camadas onde ela aprende representações de características mais complexas que são utilizadas na tarefa de classificação (KHAN et al., 2018). Um modelo geral de uma CNN, como a sequência básica ilustrada na Figura 4, possui uma única camada de entrada e de saída, juntamente com múltiplas camadas ocultas e é composto principalmente por quatro componentes: camada convolutional, camada de subamostragem, função de ativação e camada totalmente conectada, também chamada de camada de classificação (INDOLIA et al., 2018). Esse tipo de rede neural resulta em um aprendizado automático de níveis de extração de recursos úteis e a integração com suas etapas de classificações em um único *pipeline* (KHAN et al., 2018).

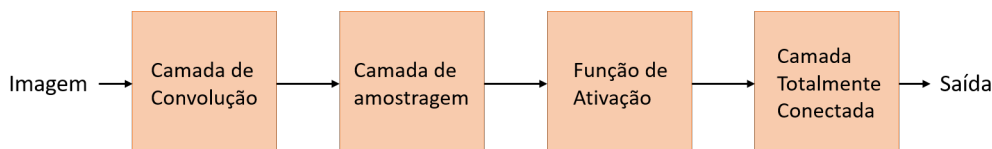


Figura 4 – Elementos básicos de uma CNN.

#### 2.3.2.1.1 Camada Convolutional

As camadas convolucionais, cuja operação de convolução foi introduzida na seção 2.2.1, são compostas por um conjunto de filtros, também chamados de *kernels*, que são convoluídos com uma dada entrada de modo a gerar um mapa de características de saída (KHAN et al., 2018). O funcionamento dessas camadas se dá pelos seguintes passos (INDOLIA et al., 2018):

- Uma imagem é fornecida como os dados de entrada da primeira camada;
- Um neurônio individual na camada seguinte é conectado a outros neurônios da camada anterior, essa correlação é chamada de campo receptivo;

- As características locais dessa imagem são extraídas usando esse campo receptivo;
- O campo receptivo de um neurônio associado a uma determinada região da camada anterior forma um vetor de peso, que permanece igual para todos os pontos dos neurônios da camada seguinte;
- Com os neurônios no plano compartilhando o mesmo peso, as características semelhantes que ocorrem em diferentes locais nos dados de entrada podem ser detectadas.

O filtro, ou *kernel*, é deslizado sobre a camada de entrada de modo a gerar o mapa de características de saída através de operações de convoluções. Por causa do campo receptivo local, o número de parâmetros treináveis é significativamente reduzido. A saída  $o_{ij}$  na próxima camada para a localização  $(i, j)$  é calculada após a aplicação da operação de convolução utilizando a Equação 2.2,

$$o_{ij} = f((K * X)_{ij} + b), \quad (2.2)$$

onde  $X$  é a entrada dada à camada,  $K$  é o filtro que desliza sobre a entrada,  $b$  é o viés,  $*$  representa a operação de convolução e  $f$  é a função de não-linearidade introduzida na rede (INDOLIA et al., 2018).

### 2.3.2.1.2 Camada de Subamostragem

A camada de convolução é seguida por uma camada de subamostragem, ou camada de subamostragem. Isso ocorre, pois a exata localização de um objeto ou característica em uma imagem de entrada se torna menos significativa após ser detectada (KHAN et al., 2018).

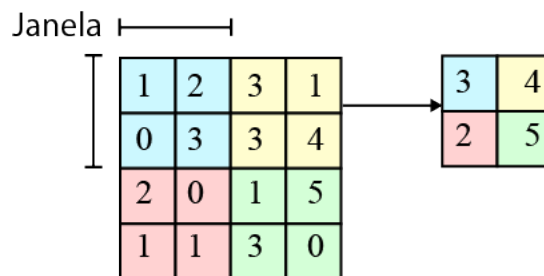


Figura 5 – operação de subamostragem máximo utilizando uma janela 2x2

A camada de subamostragem opera nos blocos do mapa de características de entrada (saída da camada anterior) e combina as ativações dessas características. Essa operação de combinação se faz definida por uma função de subamostragem, como uma função de valor médio ou de valor máximo. Assim como na camada de convolução, é preciso especificar as dimensões da janela na camada de subamostragem assim como o tamanho



do passo a ser implementado durante o deslocamento da operação sobre o mapa (KHAN et al., 2018). A vantagem de se utilizar essa técnica é a grande redução no número de parâmetros de treinamento e a aplicação da invariância de translação (LECUN et al., 1998).

### 2.3.2.1.3 Não-linearidade

Em uma CNN, as camadas com peso, assim como as camadas convolucionais, por exemplo, são frequentemente seguidas por funções de ativação não-lineares. Nas camadas totalmente conectadas essas funções são normalmente encontradas nas próprias camadas. Essas funções de ativação podem ser entendidas como um mecanismo de chave ou seleção que decide se o neurônio é ou não ativado de acordo com todas as suas entradas (KHAN et al., 2018). Na Figura 6 são mostradas algumas das funções de ativação mais comumente utilizadas para redes neurais convolucionais.

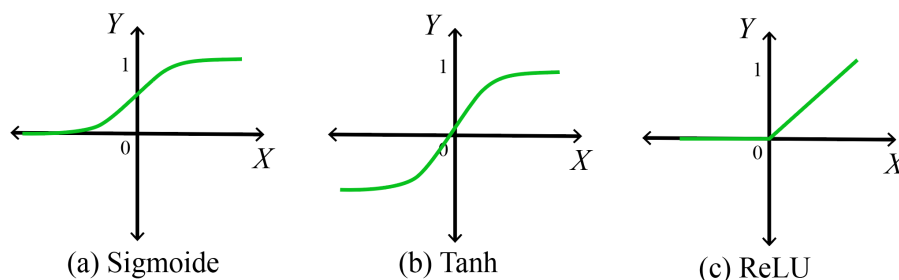


Figura 6 – Algumas das funções de ativação mais comuns que são utilizadas em redes neurais profundas.

Um grande número de referências de uso de CNN utiliza a função Sigmoid, mostrada em (a) na Figura 6, que recebe um número real como entrada e retorna um número dentro do intervalo de  $[0,1]$  (KHAN et al., 2018). Porém, em muitas das redes neurais convolucionais mais modernas a função ReLU, abreviação do inglês de *Rectified Linear Unit* mostrada em (c) na Figura 6 e na Equação 2.3,

$$ReLU(x) = \max(0, x), \quad (2.3)$$

é mais empregada devido a sua rápida computação e similaridade com os processos biológicos aos quais as redes neurais se baseiam (KHAN et al., 2018).

Outra função muito utilizada nesse contexto para classificação multi-classe é a *SoftMax*, descrita pela Equação 2.4 (REVERDY; LEONARD, 2015),

$$softmax(z)_i = \frac{e^{z_i}}{\sum_{j=1}^m e^{z_j}}, \quad (2.4)$$

cujas aplicações gera um algoritmo de aprendizado supervisionado que mapeia recursos de entrada para o plano de rótulos por meio de uma matriz de transformação linear (HAN et al., 2021).

#### 2.3.2.1.4 Camada Totalmente Conectada

Uma camada totalmente conectada geralmente é colocada após as camadas de convolução e de subamostragem ao final da arquitetura da rede. Para servir de entrada na camada totalmente conectada, a matriz passa por uma operação de achatamento, onde a saída da camada de subamostragem é convertida em uma única coluna  $x_n$ , como mostrado na Figura 7. Em sequência, cada neurônio das camadas seguintes possui conexões com todas as ativações da camada anterior, resultando em camadas totalmente conectadas (KHAN et al., 2018).

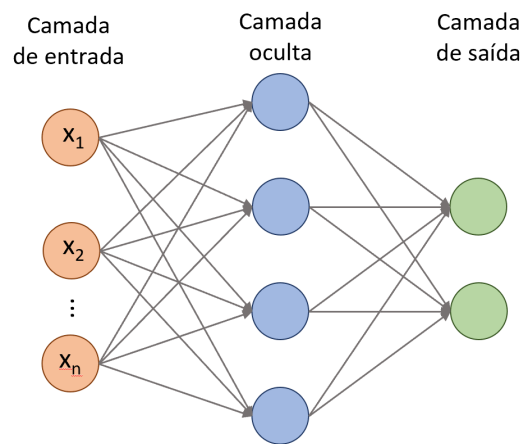


Figura 7 – Representação das conexões dos neurônios de uma camada totalmente conectada de uma CNN após o procedimento de achatamento.

#### 2.3.2.1.5 Dropout

Para fazer com que uma rede classifique objetos a partir de milhares de imagens, é necessária a implementação de uma rede de grande escala que possa suportar uma demanda intensa de treinamento e um número grande de pesos, o que leva ao problema de sobreajuste, tratado na seção 2.3.2.2.1. *Dropout*, ou abandono, é um método de regularização criado para superar esse problema. Ele realiza uma amostragem aleatória de uma sub-rede dentro da rede em grande escala, criando um efeito de conjunto durante a fase de teste onde a rede completa é utilizada para realizar a previsão (KHAN et al., 2018). Durante o treinamento, alguns neurônios são selecionados aleatoriamente para serem ignorados, de forma que suas contribuições para o restante da rede são temporariamente removidas, como é ilustrado na Figura 8. Estas remoções resultam em uma melhor generalização da rede e uma menor sensibilidade aos pesos específicos dos neurônios (SRIVASTAVA et al., 2014).

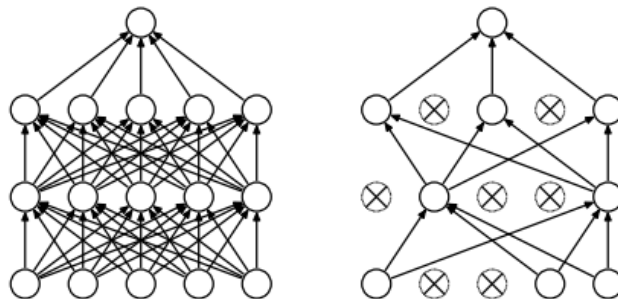


Figura 8 – Representação de aplicação de *Dropout*. À direita: um modelo de rede com duas camadas ocultas. À esquerda: um exemplo de rede com *Dropout* aplicado (SRIVASTAVA et al., 2014).

### 2.3.2.1.6 Normalização de Lote

Em 2015, Ioffe e Szegedy propuseram o conceito de Normalização de Lote como estratégia para o treinamento de redes neurais que reduz substancialmente o tempo de treinamento, elimina a necessidade de uso de *dropout*, diminui a quantidade de regularização necessária (melhorando a precisão dos dados de validação) (IOFFE; SZEGEDY, 2015).

A Normalização de Lote é utilizada para prevenir a Mudança de Covariância Interna, definida como a mudança na distribuição de ativações de cada camada conforme os parâmetros são atualizados durante o treinamento (KHAN et al., 2018). Ioffe e Szegedy avaliam que se os valores de entrada de cada camada forem estabilizados, é possível prevenir que a função de ativação atribua os valores de entrada aos valores máximo ou mínimo da função de ativação (IOFFE; SZEGEDY, 2015).

### 2.3.2.2 Treinamento

Para se inicializar o treino de uma rede neural artificial, é normalmente feita a divisão dos dados disponíveis em três conjuntos: dados de treino, dados de validação e os dados de teste, como mostrado na Figura 9. Dados de validação e de treinamento separados são necessários, pois o treinamento de um modelo envolve o ajuste fino de seus hiperparâmetros e a seleção de um modelo. Como esse processo é realizado se baseando no desempenho do conjunto de validação, o vazamento de informações de um conjunto para o outro pode gerar um problema de sobreajuste do modelo para o conjunto de validação, diminuindo o desempenho do modelo em sua avaliação com o conjunto de dados de teste (YAMASHITA; NISHIO; DO, 2018).

O conjunto de dados de treinamento é utilizado para se treinar a rede, o que se resume ao processo de se encontrar os devidos parâmetros, como os pesos das camadas totalmente conectadas e os filtros *kernels* nas camadas convolucionais, de modo a minimizar as diferenças entre as previsões de saída e as etiquetas com as respostas corretas do

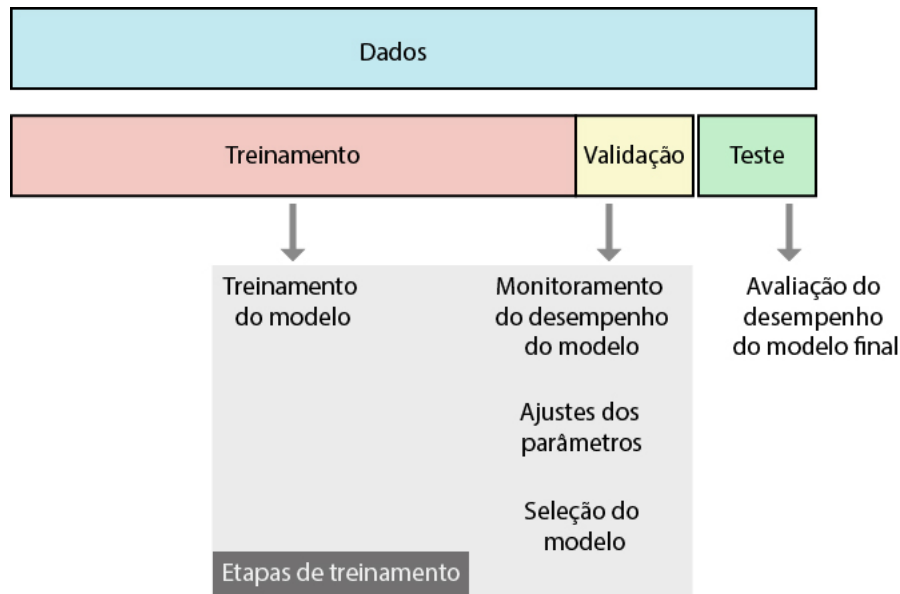


Figura 9 – Representação das etapas de treino de uma rede neural, adaptado de (YAMASHITA; NISHIO; DO, 2018).

conjunto de dados de treinamento supervisionado (YAMASHITA; NISHIO; DO, 2018)]. O método mais comumente utilizado para o treinamento dessas redes é o chamado *Back-propagation*, ou retro-propagação, onde estes parâmetros pré-inicializados são modificados para a otimização da rede (INDOLIA et al., 2018). Como geralmente se trata de uma rede de *feedforward*, esse processo se inicia computando as saídas em cada uma das camadas e calculando o componente de erro introduzido na última camada. Em seguida, para se otimizar a rede, os gradientes computados são retro-propagados. Esses passos são repetidos até que a eficácia da rede seja observada (INDOLIA et al., 2018).

#### 2.3.2.2.1 Sobreajuste

Sobreajuste, se refere a um dos principais desafios de aprendizado de máquina, onde um modelo que está aprendendo a reconhecer regularidades específicas de um conjunto de dados acaba memorizando características irrelevantes para a classificação desejada e, conseqüentemente, obtendo um desempenho menos eficaz em um determinado banco de dados. Um método de se verificar a ocorrência de sobreajuste nos dados de treino é o monitoramento das curvas de perda, como mostrado na Figura 10 e da precisão dos conjuntos de dados de treino e de validação e a comparação entre esses resultados (HINTON et al., 2012).

#### 2.3.2.2.2 *K-Fold*

A técnica de validação cruzada *k-fold* consiste em dividir a base dados de forma aleatória em um número  $k$  de conjuntos, contendo em cada um aproximadamente o mesmo

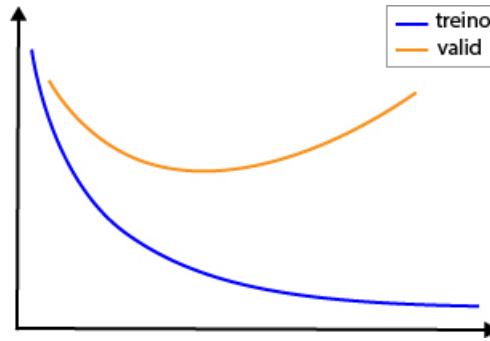


Figura 10 – Exemplo de curva de perda apresentando sobreajuste: A curva de perda dos dados de validação está muito acima e divergente da curva de perda dos dados de treino, que continua a decrescer. Na seção 2.4.2 é visto um pouco mais sobre esse tipo de curva.

número de amostras. O valor de  $k$  deve ser escolhido de maneira que os conjuntos de treino e teste representem corretamente em proporção o conjunto de dados originais sem que os dados sejam repetidos dentro dos conjuntos. A cada ciclo de treinamento e teste, um desses conjuntos ( $k=1$ ) é selecionado para ser o conjunto de dados de teste enquanto os demais são utilizados para treino do modelo, como mostrado na Figura 11. As métricas de desempenho (que serão explicadas mais à frente) geral do  $k$ -fold são calculadas a partir das médias das métricas obtidas em cada ciclo de *fold* (BERRAR, 2018).

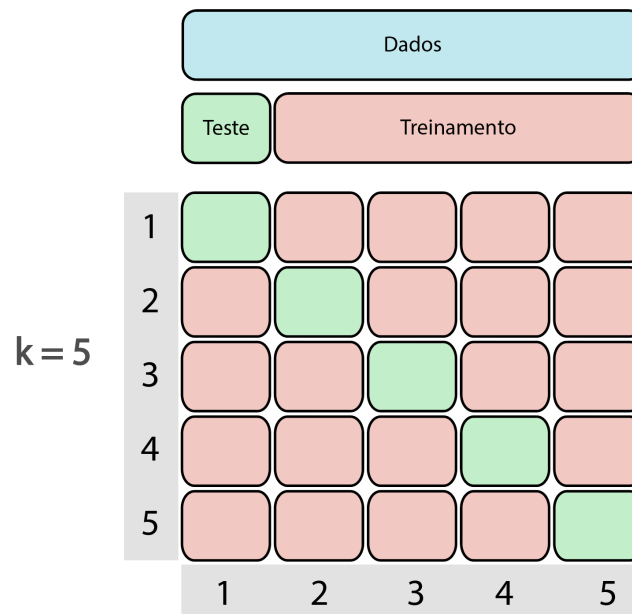


Figura 11 – Ilustração da aplicação do método  $k$ -fold com  $k=5$ .

### 2.3.2.2.3 Função de Perda

A função de perda de uma Rede Neural convolucional é utilizada durante o processo de treinamento da máquina. Ela realiza uma estimativa da qualidade das previsões feitas

pela rede nos dados de treino, os quais os resultados de previsão são conhecidos. Algumas funções de perda mais utilizadas são a de Erro Médio Quadrático - MSE, do inglês *Mean Squared Error* e a de Entropia Cruzada - CE, do inglês *Cross-Entropy loss* (KHAN et al., 2018). A MSE é mais utilizada para casos de regressão, onde o objetivo é realizar a previsão de um valor contínuo baseado em um número de parâmetros, enquanto a CE é mais utilizada para casos de classificação.

A função de perda de Entropia Cruzada é dada pela Equação 2.5 abaixo,

$$Perda(p, y) = - \sum_{i=1}^c y_i \cdot \log(p_i), \quad (2.5)$$

onde  $c$  é o número de classes,  $y_i$  é o valor real da classe de comparação e  $p_i$  é o valor previsto para essa classe através da função de ativação. Ela possui algumas variações, duas das quais são a Entropia Cruzada Categórica - CCE, do inglês *Categorical Cross-Entropy* e a Entropia Cruzada Binária - BCE *Binary Cross-Entropy*. A diferença entre elas se dá pela aplicação e pela função de ativação utilizada. Na CCE é utilizada uma função de ativação *softmax* seguida de uma CE, e é utilizada para classificação de múltiplas classes. Já a BCE é composta por uma função sigmoide seguida de uma CE, utilizada para o caso de classificação binária.

Para se minimizar a função de perda, uma das ferramentas mais utilizadas é o Método da Descida do Gradiente, também chamado de Gradiente Descendente. Esse método é aplicado para diminuir a diferença entre a saída e as respostas originais durante o treino. Os resultados previstos dependem dos parâmetros da rede, com isso, o gradiente é descrito matematicamente como uma derivada parcial da perda em relação a cada parâmetro que pode ser aprendido, reduzindo a função progressivamente a um valor mínimo a cada passo (KHAN et al., 2018).

#### 2.3.2.2.4 Regularização

Técnicas de regularização são aplicadas em modelos de redes neurais com o intuito de se evitar a ocorrência de sobreajuste. A Regularização L2, se trata de uma das técnicas de regularização mais comumente utilizadas, ela é aplicada modo que a função de perda seja minimizada e uma penalidade na norma  $L_2$  dos pesos seja adicionada, como mostrado na Equação 2.6,

$$C = C_0 + \frac{\lambda}{2n} \sum_w w^2 \quad (2.6)$$

onde  $C_0$  é a função de perda não-regularizada,  $C$  é a função de perda regularizada e  $\lambda$  é o parâmetro da regularização, diretamente proporcional à quantidade de regularização aplicada.

### 2.3.2.2.5 RandAugment

*RandAugment*, ou Aumento Aleatório, é uma técnica utilizada para se aumentar a quantidade de dados de um banco de maneira automática. Nela se tem 2 hiperparâmetros: M e N, onde N é o número de transformações de aumento a serem aplicadas sequencialmente e M é a magnitude de todas as essas transformações. Para se diminuir o espaço de parâmetros enquanto se mantém uma diversidade de imagens, os métodos e probabilidades aprendidos para cada transformação são substituídos pelo procedimento de sempre selecionar uma transformação com uma probabilidade uniforme  $\frac{1}{K}$ , onde K é o número de opções de transformação. Com isso, dadas N transformações para uma determinada imagem de treino, o aumento aleatório pode expressar KN métodos potenciais. Dentro destas transformações estão autocontraste, rotação em x e em y, translação em x e em y, mudanças de brilho, nitidez contraste entre outras (CUBUK et al., 2019).

### 2.3.2.3 Arquiteturas de CNNs

Nos últimos anos foram desenvolvidas várias arquiteturas de CNNs, cada uma visando suprir alguma necessidade ou melhorar em algum aspecto em relação às anteriores. Abaixo são apresentadas algumas das mais utilizadas.

#### 2.3.2.3.1 VGGNet

VGGNet, ou somente VGG, do inglês *Visual Geometry Group* é uma arquitetura de CNN proposta por A. Zisserman and K. Simonyan da Universidade de Oxford (SIMONYAN; ZISSERMAN, 2015), introduzida em 2014. Ela se tornou bastante popular devido à sua simplicidade de modelo e pelo uso de filtros convolucionais de pequeno porte, que levam à composição de redes mais profundas (KHAN et al., 2018).

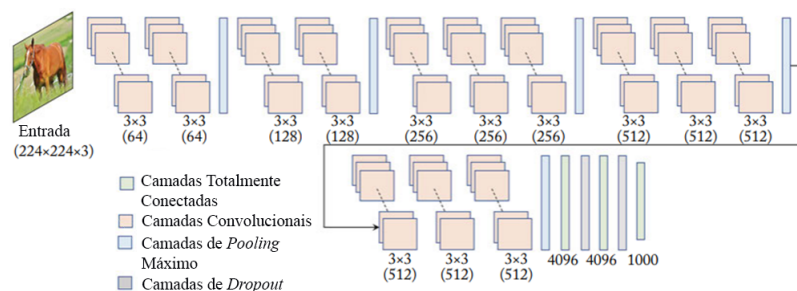


Figura 12 – Representação da configuração da arquitetura VGGNet modelo VGG16 (KHAN et al., 2018)

Os modelos mais comuns da VGGNet são o VGG16 e o VGG19, onde o “16” e o “19” correspondem à quantidade de camadas suportadas pela arquitetura. Nesses modelos os filtros convolucionais, geralmente maiores, são substituídos por filtros de tamanho 3x3,

colocados um após o outro, aprofundando a rede (SIMONYAN; ZISSERMAN, 2015). Uma das grandes contribuições da VGGNet foi mostrar a relação entre profundidade e desempenho das redes Neurais.

### 2.3.2.3.2 ResNet

A ResNet, do inglês *Residual Network*, foi a vencedora do ImageNet de 2015 com uma grande diferença de desempenho, reduzindo a taxa de erro dos 5 primeiros colocados para aproximadamente 3,6% em comparação com o vencedor de desempenho do ano anterior, que foi a GoogleNet com 6,7% (KHAN et al., 2018).

A primeira ResNet proposta foi a ResNet-34, que se baseia em transformar uma rede simples de 34 camadas de parâmetros em seu equivalente de redes residuais. Para esse modelo, a rede simples foi inspirada na VGGNet com suas camadas convolucionais utilizando filtros 3x3. Comparando com as VGGNets, as ResNets possuem menor complexidade e menor número de filtros, de modo que o modelo VGG19, com 19 camadas de parâmetros, realiza 19.6 bilhões de FLOPs, enquanto a ResNet com 34 camadas de parâmetros realiza 3.6 bilhões de FLOPs (HE et al., 2015).

A aplicação da ResNet também é uma solução ao problema da dissipação do gradiente, onde à medida que mais camadas usando certas funções de ativação são adicionadas às redes neurais, os gradientes da função de perda se aproximam de zero, tornando a rede difícil de treinar. Ela contorna esse problema conforme é representado no exemplo da Figura 13, onde o valor do início do bloco  $x$  é adicionado diretamente ao final do bloco ( $F(x)+x$ ) por uma conexão residual, também chamada de conexão de salto. Essa conexão não passa pelas funções de ativação que achatariam as derivadas, resultando em um bloco com uma derivada geral mais elevada (HE et al., 2015). Na Figura 14 é mostrado um exemplo de arquitetura ResNet, mostrando suas camadas de convolução, camadas de subamostragem, camadas totalmente conectadas e função de ativação.

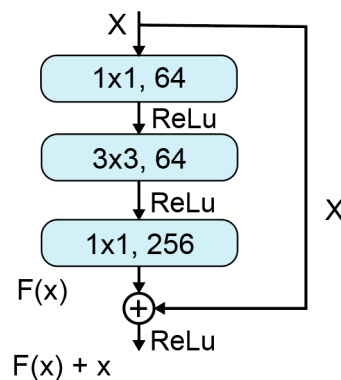


Figura 13 – Exemplo de um bloco residual ResNet.



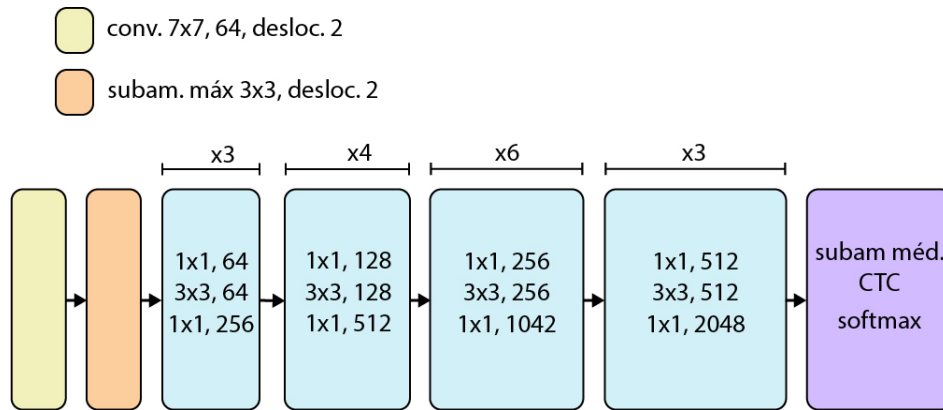


Figura 14 – Representação da arquitetura da ResNet-50, onde os blocos em azul contêm as convoluções dos blocos residuais e CTC representa as camadas totalmente conectadas.

### 2.3.2.3.3 EfficientNet

A mudança de escala de uma arquitetura de rede neural é uma estratégia bastante utilizada para os mais diversos objetivos. Um exemplo disso são os modelos da ResNet, que podem ser escalonados da ResNet18 à ResNet200. Em 2019, Mingxing Tan e Quoc V. Le propuseram um modelo de arquitetura de rede com um coeficiente composto simples, mas altamente eficaz para escalar CNNs de uma forma mais estruturada (TAN; LE, 2020).

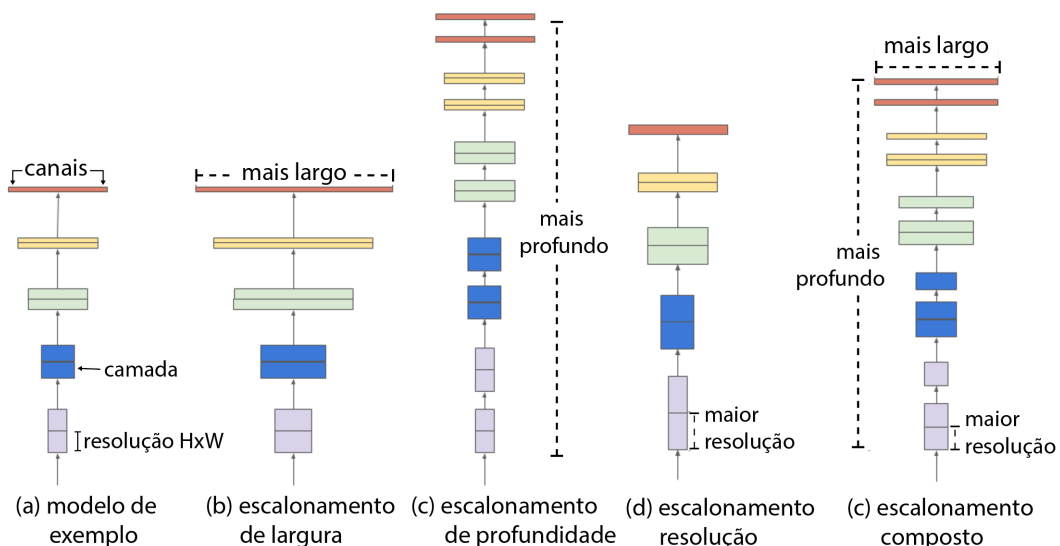


Figura 15 – Comparação de diferentes métodos de escalonamento de redes neurais, adaptado de (TAN; LE, 2020).

Esse modelo de escalonamento composto, mostrado na Figura 15, pode ser utilizado de maneira geral para outras arquiteturas. No modelo da EfficientNet-B0, a primeira arquitetura desse modelo que serviu de base para as demais EfficientNet, os autores uti-

lizam o Bloco Residual Invertido como bloco principal (TAN; LE, 2021). De maneira oposta ao bloco residual da ResNet, ele possui uma estrutura estreita->larga->estreita, onde primeiro amplia-se com uma convolução 1x1, depois usa-se uma convolução profunda 3x3, o que reduz bastante a quantidade de parâmetros e, em seguida, usa-se novamente uma convolução 1x1, para reduzir o número de canais de modo que a entrada e a saída possam ser adicionadas, como representado na Figura 16 (SANDLER et al., 2019). Em uma convolução profunda, a imagem e o filtro são separados em diferentes canais, em seguida é realizada a convolução da imagem com o canal correspondente e os resultados são empilhados juntos novamente, necessitando de menos parâmetros do que uma convolução normal. No bloco residual invertido também é utilizada uma função de Aperto e Agitação que refaz os pesos dos canais por meio de interações entre esses canais fazendo a subamostragem média de todo o mapa de características (HU et al., 2019).

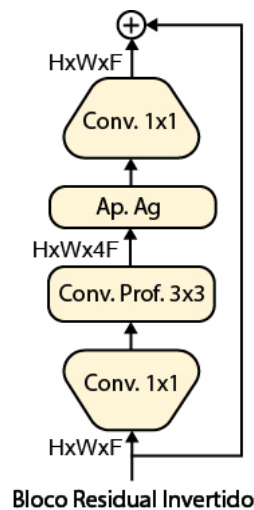


Figura 16 – Representação do bloco residual invertido, onde H,W e F representam as dimensões dos dados em questão, H representa a altura, W a largura e F a profundidade.

Outra característica da EfficientNet é o uso da função de ativação *Swish*, que é a multiplicação de uma função linear  $x$  e uma sigmoide  $s$ , como mostrado na Equação 2.7,

$$Swish(x) = x * s(x). \quad (2.7)$$

#### 2.3.2.3.4 ResNet-RS

A ResNet-RS é uma família de arquiteturas ResNet que possuem melhoras significativas em velocidade de TPU - Unidade de Processamento de Tensor, do inglês *Tensor Processing Unit*. Essa velocidade de TPU atingida é entre 1.7 a 2.7 vezes mais rápida do que as EfficientNets, porém com uma precisão semelhante no ImageNet quando não são utilizados dados extras de treino. Nela são apresentadas estratégias envolvendo escalona-

mento em profundidade do modelo em escala em regimes onde pode ocorrer sobreajuste (BELLO et al., 2021).

Nesse modelo de arquitetura também é implementado o *RandAugment* para se aumentar do número de dados do banco e são aplicadas, para a regularização, técnicas como regularização L2, *dropout* e profundidade estocástica. Profundidade estocástica é uma técnica que visa diminuir a profundidade de uma rede apenas durante o treinamento, mantendo-a inalterada durante o teste final. Isso é atingido descartando aleatoriamente blocos residuais inteiros durante o treinamento e contornando suas transformações através de conexões de salto. (HUANG et al., 2016).

Algumas mudanças importantes na arquitetura da ResNets para a criação das ResNets-RS foram a implementação de funções *Squeeze-and-Excitation* em todos os blocos de gargalo e a adição de características da ResNet-D. A segunda se trata de uma combinação de determinados ajustes na arquitetura geral, como substituição de blocos de convolução maiores por vários menores, mudanças em tamanhos de passadas em convoluções em determinados blocos, etc. A arquitetura resultante, ResNet-Rs, pode ser observada na Figura 17 (BELLO et al., 2021).

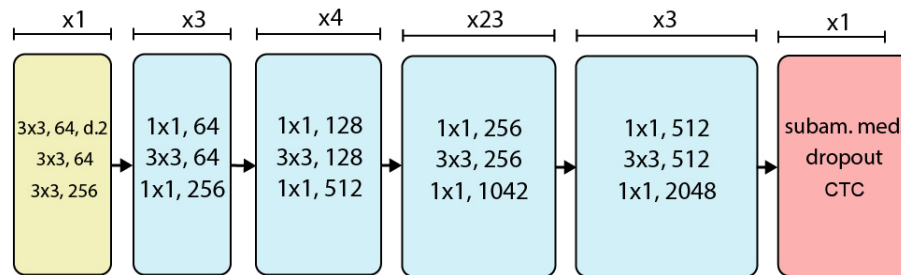


Figura 17 – Ilustração de uma arquitetura ResNet-RS

## 2.4 Ferramentas de Validação

### 2.4.1 Métricas de Desempenho

Em uma classificação binária, cada entrada é assimilada a uma ou duas classes, geralmente se utilizando rótulos de 1 ou 0, ou positivo e negativo. Já em uma classificação de múltiplas classes, os rótulos vão de 0 a  $n$ , sendo  $n$  igual ao o número de rótulos menos 1. Para se avaliar o desempenho de uma rede neural durante seu treinamento, é utilizada uma métrica chamada Matriz de Confusão, que ajuda a visualizar a relação entre os valores verdadeiros dos rótulos dos dados de treino e os valores obtidos na previsão realizada pela rede. Essas relações são mostradas na Figura 18, onde se pode observar a organização nos seguintes conjuntos (GIANEY; CHOUDHARY, 2017):

- **Positivo Verdadeiro (TP - *True Positive*)**: O dado obtido da previsão é verdadeiro, correspondente com o rótulo.
- **Falso Positivo (FP - *False Positive*)**: O dado obtido da previsão é positivo, porém, de acordo com o rótulo, era para ser negativo.
- **Negativo Verdadeiro (TN - *True Negative*)**: O dado obtido da previsão é negativo, correspondente com o rótulo.
- **Falso Negativo (FN - *False Negative*)**: O dado obtido da previsão é negativo, porém, de acordo com o rótulo, era para ser positivo.

Classe Verdadeira	0	TP	FN
	1	FP	TN
		0	1
		Classe Prevista	

Figura 18 – Representação de uma matriz de confusão com os valores focados na classe "0", organizado em Positivo Verdadeiro, Positivo Falso, Negativo Falso e Falso Verdadeiro.

Para uma rede neural desempenhando classificação de múltiplas classes, os valores positivos e negativos são obtidos de maneira similar, porém, com o aumento da matriz, mais valores de verdadeiro e falso são adicionados. Como consequência, para uma classificação de  $n$  classes, tem-se que TP é o valor de intersecção da coluna com a linha da classe em foco. Os valores de TN, FP e FN são iguais à soma de todos os seus valores para essa mesma classe, ou seja,  $TN = TN_1 + TN_2 + \dots + TN_n$ ,  $FP = FP_1 + FP_2 + \dots + FP_n$  e  $FN = FN_1 + FN_2 + \dots + FN_n$  para a classe em questão. Essas relações podem ser verificadas na Figura 19.

A partir das relações previstas nas matrizes de confusão apresentadas, é possível obter valores de determinadas métricas de desempenho, como acurácia, precisão e sensibilidade, especificidade e a *F1-Score*.

O conceito de acurácia se dá pela métrica que descreve o desempenho do modelo para todas as classes. A acurácia é calculada como a razão entre o número de previsões verdadeiras e o número total de previsões, como mostrado na Equação 2.8,

$$\text{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN}. \quad (2.8)$$

Classe Verdadeira	A	TP	FN	FN
	B	FP	TN	TN
	C	FP	TN	TN
		A	B	C

(A)

Classe Verdadeira	A	TN	FP	TN
	B	FN	TP	FN
	C	TN	FP	TN
		A	B	C

(B)

Classe Verdadeira	A	TN	TN	FP
	B	TN	TN	FP
	C	FN	FN	TP
		A	B	C

(C)

Figura 19 – Representação de uma matriz de confusão para a classificação das classes A, B e C, onde (A), (B) e (C) possuem os valores Positivo Verdadeiro, Positivo Falso, Negativo Falso e Falso Verdadeiro focados em suas respectivas classes.

A precisão descreve na confiabilidade do modelo para a classificação dos dados como positivo. Ela é calculada como o número de positivos verdadeiros sobre o número total de positivos obtidos, como mostrada na Equação 2.9,

$$Precisão = \frac{TP}{TP + FP}. \quad (2.9)$$

A sensibilidade, também chamada de revocação, ou *recall*, mede a capacidade do modelo de detectar os dados positivos. Ela é calculada como o número de positivos verdadeiros sobre o valor da soma dos números de positivos verdadeiros mais os falsos negativos, como mostrado na Equação 2.10,

$$Sensibilidade = \frac{TP}{TP + FN}. \quad (2.10)$$

A especificidade se resume como a razão entre os casos verdadeiros negativos e a soma dos falsos positivos com os verdadeiros positivos, ou seja, todos os casos negativos. Visualizando a Equação 2.11,

$$Especificidade = \frac{TN}{TN + FP}. \quad (2.11)$$

pode-se perceber que a especificidade é uma análise oposta à da sensibilidade.

A escolha entre se utilizar a sensibilidade ou a precisão como métrica de avaliação do modelo se dá pelo seu objetivo principal desejado, priorizando a detecção de Falsos Positivos ou Falsos Negativos. Para o trabalho em questão, como não existem análises prévias de redes neurais para o banco de imagens utilizado, o uso de ambas as métricas poderia trazer uma melhor análise de resultados do modelo. Porém na prática, é comum a diminuição da sensibilidade do modelo ao se tentar aumentar sua precisão e vice-versa. Uma alternativa de análise existente é o uso da *F1-Score*, descrita como a média harmônica entre a precisão e a sensibilidade como mostrado na Equação 2.12,

$$F1-Score = \frac{2}{\frac{1}{Sensibilidade} + \frac{1}{Precisão}}. \quad (2.12)$$

## 2.4.2 Curvas de Aprendizado

O termo "curva de aprendizado" se refere à curva que descreve o desempenho do sistema em relação a algum tipo de recurso limitado utilizado para resolver uma tarefa. O desempenho é uma medida que mostra o quão bem essa tarefa é resolvida. Essas curvas são amplamente utilizadas na área de aprendizado de máquina para tomar decisões quanto à aquisição de dados, à melhor seleção entre modelos de treinamento e quanto à parada precoce do treinamento (MOHR; RIJN, 2022; PERLICH, 2011). A parada precoce é uma técnica utilizada para se evitar o sobreajuste no treinamento, ela pode ser aplicada diminuindo o número de iterações de treino da máquina de modo que o treinamento termine antes da máquina começar a memorizar grandes quantidades de parâmetros irrelevantes, como mostrado na Figura 20.

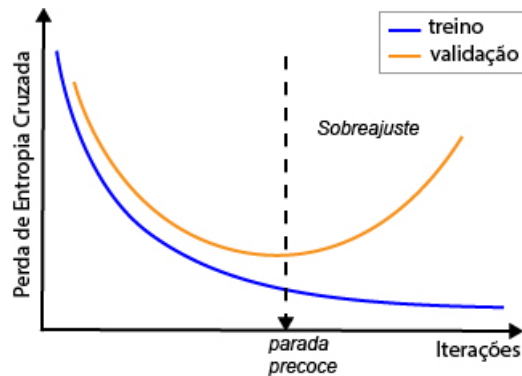


Figura 20 – Exemplo de curvas de perda de validação e treino

As curvas de aprendizado podem ser comumente observadas de dois pontos de vista: otimização, como as curvas de perda mostradas na Figura 20 e desempenho, como a curva de acurácia mostrada na Figura 21 (WISTUBA; PEDAPATI, 2020; ANZANELLO; FOGLIATTO, 2011; MOHR; RIJN, 2022). Uma estratégia comum para problemas de classificação em aprendizado de máquina é o uso de curvas de aprendizado para diferentes métricas. Nestes casos são geradas duas curvas, uma para os dados de treino e outra para os dados de validação (ANZANELLO; FOGLIATTO, 2011).

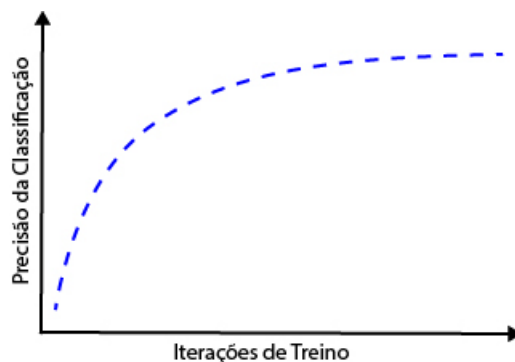


Figura 21 – Exemplo de curva de aprendizado de acurácia

## 3 Metodologia

### 3.1 Banco de Dados

O banco de dados utilizado no trabalho se trata de um banco de imagens desenvolvido pelo Laboratório de Nanobiotecnologia da Embrapa Recursos Genéticos e Biotecnologia, disponibilizado para a realização deste trabalho, constituído de fotos de superfície de folhas em nível microscópico, capturadas com o auxílio de um microscópio portátil para *smartphone* Go Micro. Todas as imagens foram obtidas em condições próximas ou semelhantes de iluminação e distância focal, possuem dimensões de 1936x1936 pixels, uma resolução de 72 dpi e padrão de cores sRGB. O banco possui um total de 1040 imagens, 52 para cada espécie. Na Figura 22 são mostradas as divisões de conjuntos implementadas nas imagens do banco, agrupadas em família, com cada imagem representando uma espécie dessa família, e as famílias agrupadas em acumuladoras e excludentes. Vale lembrar que, para o banco de dados utilizado, cada família está sendo rotulada como “Acumuladora” ou “Excludente” por causa da natureza das espécies utilizadas no trabalho em questão, o que não é uma regra, já que uma mesma família de plantas pode apresentar membros acumuladores e excludentes dependendo da espécie.

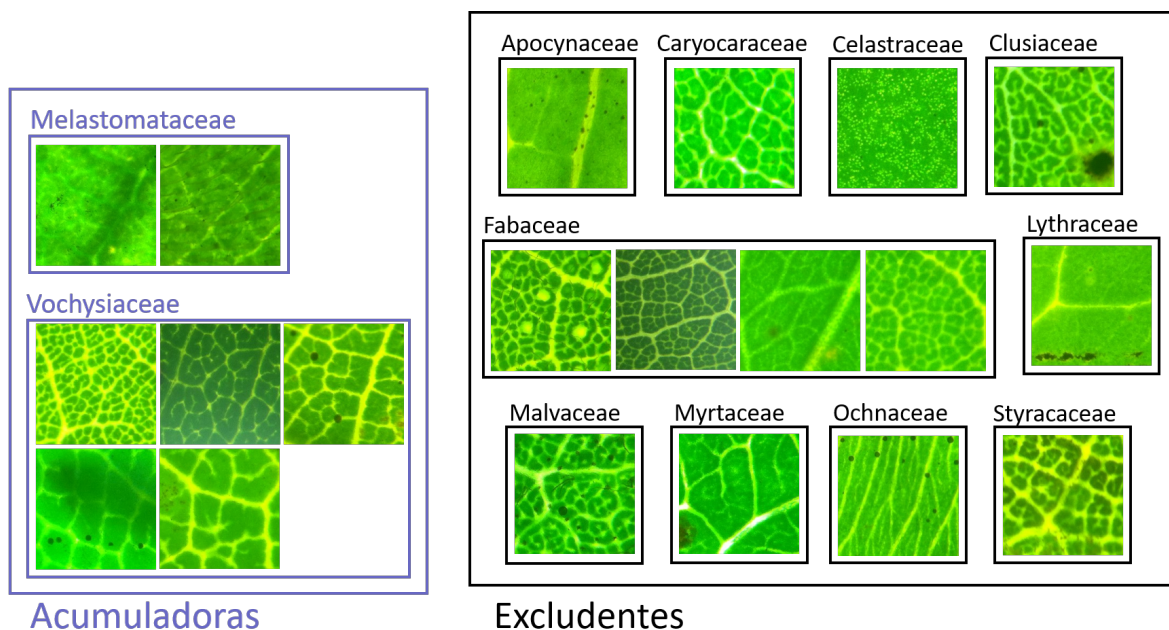


Figura 22 – Imagens retiradas do banco de dados. Cada figura representa uma das espécies de plantas do banco de dados, essas espécies são agrupadas em famílias e as famílias, agrupadas de acordo com o mecanismo de resistência ao alumínio.

As folhas em questão pertencem a determinadas plantas do cerrado que possuem

resistência à presença de alumínio no solo, podendo ser classificadas quanto ao seu mecanismo de resistência. A relação da quantidade de imagens por tipo pode ser verificada na Tabela 1. A relação entre família e método de resistência de cada espécie dessas folhas pode ser verificada na Tabela 2, adaptada de (OLIVEIRA et al., 2019).

	Acumuladoras (A)	Excludentes (E)	Total
Famílias	2	10	12
Espécies	7	13	20
<b>Número de Imagens</b>	364	676	<b>1040</b>

Tabela 1 – Tabela da relação da quantidade de imagens, de famílias e de espécies para cada mecanismo de resistência.

Mecanismo de Resistência	Família	Espécie
A	Melastomataceae	<i>Miconia ferruginata</i>
A	Melastomataceae	<i>Pleroma stenocarpum</i>
A	Vochysiaceae	<i>Qualea parviflora</i>
A	Vochysiaceae	<i>Vochysia rufa</i>
A	Vochysiaceae	<i>Qualea grandiflora</i>
A	Vochysiaceae	<i>Vochysia elliptica</i>
A	Vochysiaceae	<i>Salvertia convallariodora</i>
E	Apocynaceae	<i>Hancornia speciosa</i>
E	Caryocaraceae	<i>Caryocar brasiliense</i>
E	Celastraceae	<i>Salacia crassifolia</i>
E	Clusiaceae	<i>Kielmeyera coriacea</i>
E	Fabaceae	<i>Tachigali subvelutina</i>
E	Fabaceae	<i>Hymenaea stigonocarpa</i>
E	Fabaceae	<i>Stryphnodendron adstringens</i>
E	Fabaceae	<i>Plathymenia reticulata</i>
E	Lythraceae	<i>Lafoensia pacari</i>
E	Malvaceae	<i>Eriotheca gracilipes</i>
E	Myrtaceae	<i>Psidium myrsinoides</i>
E	Ochnaceae	<i>Ouratea hexasperma</i>
E	Styracaceae	<i>Styrax ferrugineus</i>

Tabela 2 – Relação entre família e mecanismo de resistência correspondente para cada espécie de planta, onde A = Acumuladora e E = Excludente, (OLIVEIRA et al., 2019) cujas imagens das folhas estão sendo utilizadas no banco.

Neste trabalho foram aplicados dois conjuntos de classificação distintos para o mesmo banco de dados: classificação por família (com 12 classes) e classificação binária entre acumuladora e excludente. Para a aplicação no programa, o banco de dados foi dividido em aproximadamente 935 imagens para o processo de treinamento e 104 imagens para os testes. No processo de treinamento, as 935 imagens são distribuídas de forma automática em aproximadamente 841 imagens para teste e 94 para validação.



Lote, no contexto de aprendizado de máquina, é um termo que se refere ao conjunto de imagens utilizado em uma iteração. Para este trabalho, foi escolhido um tamanho de lote de 52 imagens, de modo que proporcionalmente fossem distribuídas entre 2 a 3 imagens de cada família por lote, visto que a quantidade de imagens por família é desbalanceada, como mostrado na Tabela 3. Enquanto o teste de cada *fold* é realizado com 2 lotes de imagens, o treino e a validação possuem as imagens distribuídas em 17 e 2 lotes, respectivamente. Com isso, a cada 8 iterações de treino, uma iteração de validação é realizada.

Família	Quantidade de imagens	Proporção	Imagens por lote
Apocynaceae	52	5%	2,6
Caryocaraceae	52	5%	2,6
Celastraceae	52	5%	2,6
Clusiaceae	52	5%	2,6
Fabaceae	208	20%	10,4
Lythraceae	52	5%	2,6
Malvaceae	52	5%	2,6
Melastomataceae	104	10%	5,2
Myrtaceae	52	5%	2,6
Ochnaceae	52	5%	2,6
Styracaceae	52	5%	2,6
Vochysiaceae	260	25%	13

Tabela 3 – Proporção da quantidade de imagens do banco de dados para cada família de planta, com o número de imagens por lote sendo representado por uma média da quantidade de imagens por lote de cada família de acordo com a proporção.

## 3.2 Arquiteturas Utilizadas

### 3.2.1 ResNet-RS

Em 2019 foi proposto por Ekin D. Cubuk o uso do *RandAugment* para o treino da EfficientNet, o que melhorava seu desempenho no ImageNet, com a EfficientNet-B7 chegando a atingir aproximadamente 85% sem o uso de dados extras de treino. Comparando esse desempenho com a da ResNet-RS, conforme a Tabela 4, percebe-se que apesar de utilizar um número maior de parâmetros, os modelos de ResNet-RS consomem uma quantidade muito menor de memória e possuem uma maior velocidade em TPUs e GPUs enquanto obtêm valores de precisão muito próximos. A Latência de TPU é traduzida como o tempo por passo de treinamento para 1024 imagens em 8 núcleos TPUv3. A memória é traduzida em 32 imagens por núcleo, utilizando precisão *bfloat16* no TensorFlow 1.x (BELLO et al., 2021).

Modelo	Imagem (pixels)	Params (M)	FLOPs (B)	V100 Lat. (s)	TPUv3 Lat. (ms)	Precisão (%)
EfficientNet-B0	224	5.3	0.8	0.47	90	77.1
EfficientNet-B1	240	7.8	1.4	0.82	150	79.1
ResNet-RS-50	160	36	4.6	0.31	70	78.8
EfficientNet-B2	260	9.2	2.0	1.03	210	80.1
ResNet-RS-101	160	64	8.4	0.48	120	80.3
EfficientNet-B3	300	12	3.6	1.76	340	81.6
ResNet-RS-101	192	64	12	0.70	170	81.2
ResNet-RS-152	192	87	18	0.99	240	82.0
EfficientNet-B4	380	19	8.4	4.0	710	82.9
ResNet-RS-152	224	87	24	1.48	320	82.8
ResNet-RS-152	256	87	31	1.76	410	83.0
EfficientNet-B5	456	30	20	8.16	1510	83.7
ResNet-RS-200	256	93	40	2.86	570	83.4
ResNet-RS-270	256	130	54	3.76	780	83.8
EfficientNet-B6	528	43	38	15.7	3010	84.0
ResNet-RS-350	256	164	69	4.72	1100	84.0
EfficientNet-B7	600	66	74	29.9	6020	84.7
ResNet-RS-350	320	164	107	8.48	1630	84.2
ResNet-RS-420	320	192	128	10.16	2090	84.4

Tabela 4 – Comparação entre os modelos de ResNet-RS, e os modelos das EfficientNets nos quesitos de Parâmetros, FLOPs, Latência de TPU, Latência de GPU, e Precisão atingida no ImageNet sem o uso de dados extras de treino, adaptado de (BELLO et al., 2021)

Dentre os modelos de ResNets-RS apresentados na Tabela 4, foram utilizados os modelos ResNet-RS-50, ResNet-RS-101 e ResNet-RS-152, focando no menor consumo de recursos computacionais e visando na acessibilidade do projeto aos recursos gratuitos disponíveis para aprendizado de máquina. Os parâmetros utilizados nos modelos pré-treinados são baixados do repositório oficial da ResNet-RS<sup>1</sup>. Os modelos pré-treinados do repositório foram obtidos do treino dos modelos para o banco de dados da ImageNet, que possui mais de um milhão de imagens, com mil classes distintas.

### 3.3 Pré-processamento

Como as imagens do banco em questão possuíam grandes dimensões, para se diminuir o tempo de execução do programa e o gasto de recursos computacionais, foi necessário realizar um redimensionamento das imagens. Seguindo a estratégia utilizada em

<sup>1</sup> Repositório oficial da ResNet-RS: <[https://github.com/tensorflow/tpu/tree/bee9c4f6/models/official/resnet/resnet\\_rs](https://github.com/tensorflow/tpu/tree/bee9c4f6/models/official/resnet/resnet_rs)>, último acesso: 3 de maio de 2022.

outros modelos de aprendizado de máquina (YUAN et al., 2021), todas as imagens do banco foram redimensionadas para 224x224 pixels. Com essa estratégia, apesar das imagens apresentarem uma menor qualidade geral, suas características relevantes ainda são mantidas, conforme mostrado na Figura 23, enquanto menos recursos computacionais são consumidos, otimizando bastante o tempo total de treinamento.

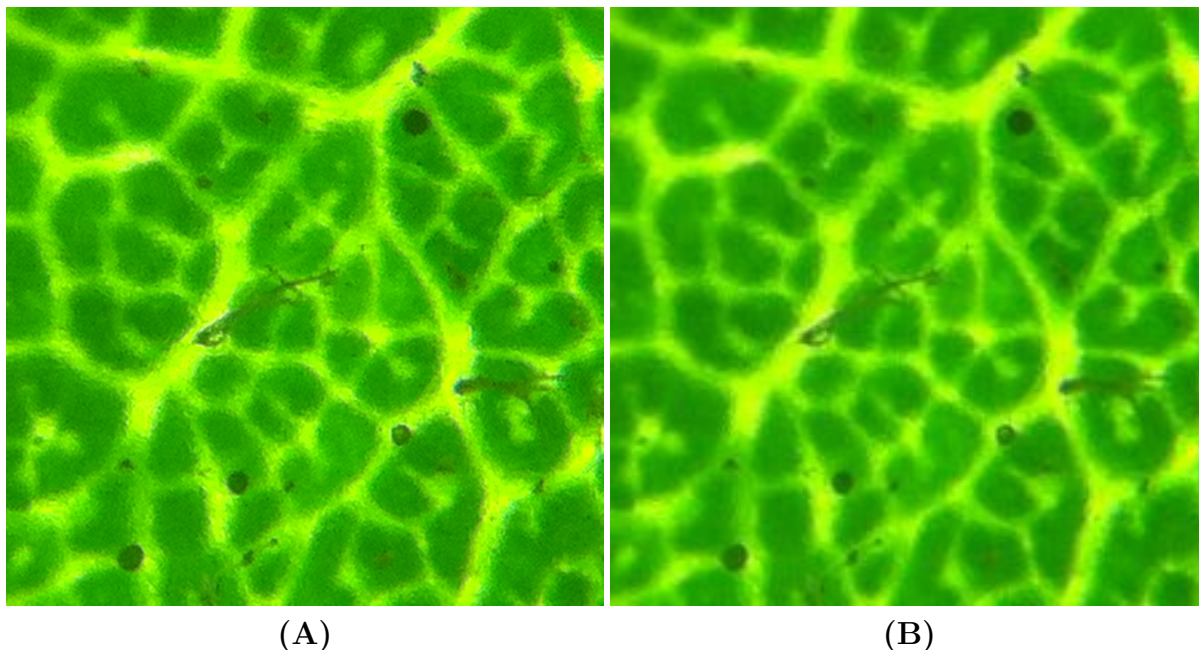


Figura 23 – (A) - Imagem de resolução original 1936x1936 pixels (reescalada). (B) - Mesma Imagem de (A), porém com 224x224 pixels de resolução e escala original.

## 3.4 Recursos Computacionais

### 3.4.1 Hardware

Para uma melhor implementação deste projeto foi necessário o uso de serviços de processamentos que permitam rodar códigos de aprendizado de máquina com agilidade e qualidade. Para isso foram estudados alguns serviços de execução de *notebooks*, dos quais foram selecionados o Kaggle<sup>2</sup> e o Google *Colaboratory*<sup>3</sup>. *Notebooks* são documentos virtuais que permitem a execução e documentação de códigos. Kaggle é uma plataforma que possibilita o uso do Notebook em um ambiente virtual personalizável com acesso a recursos computacionais voltados para execução de programas de aprendizado de máquina em nuvem. Essa plataforma interage com vários *datasets* utilizados para aprendizado de máquina em competições além de possuir um grande repositório de dados e códigos

<sup>2</sup> Site oficial da plataforma Kaggle: <<https://www.kaggle.com/>>, último acesso: 2 de maio de 2022.

<sup>3</sup> Site oficial do Google *Colaboratory*: <<https://colab.research.google.com/>>, último acesso: 27 de abril de 2022.

publicados por membros de sua comunidade. Colab, como é também chamado o Google *Colaboratory*, é um serviço igualmente voltado para a criação, execução e armazenamento de *notebooks* em nuvem (DERAKHSHAN et al., 2020; WANG et al., 2021).

Na Tabela 5 é possível fazer comparações entre as plataformas. O Google Colab possui algumas versões pagas, o Google Colab Pro e o Google Colab Pro +, que oferecem possibilidades de aumento de memória, melhoras de GPU e aumento de tempo de execução. Na tabela são mostrados alguns desses aspectos do Google Colab Pro em comparação com sua versão gratuita. O GPU utilizado pelo Google Colab Pro não é selecionável pelo usuário, sendo ajustado conforme a necessidade do programa. Vale adicionar que, além dos dados mostrados na tabela, o Kaggle também possui 20GB de espaço de disco disponível para dados salvos automaticamente. Outro aspecto importante é o uso de TPUs - *Tensor Processing Units*, disponíveis para ambos os casos, usados para acelerar as cargas de trabalho de aprendizado de máquina.

Para a execução do projeto, ambas plataformas foram testadas porém, a plataforma Kaggle foi preferida devido à facilidade da execução online do programa em segundo plano, permitindo uma melhor otimização do tempo de trabalho, função que é disponibilizada pelo Colab apenas em sua versão paga. Outros fatores que influenciaram a preferência pelo Kaggle envolvem experiência de usuário na plataforma, como a apresentação de uma interface mais intuitiva.

	Parâmetro	Google Colab - Pro	Kaggle
Geral	Máx. de execução/sessão	12 horas - 24 horas	9 horas
	Tempo ocioso	90 min	60 min
	Limite de uso de execução	Ilimitado	30 horas/semana
	Execução em segundo plano	Não - Sim	Sim
CPU	Modelo	Intel(R) Xeon(R)	Intel(R) Xeon(R)
	Frequência	2.3GHZ	2.0GHz
	Qt. de núcleos	2	4
	RAM	12GB - 26.75GB	16GB
	Espaço de disco	25GB	5GB
GPU	Modelo	K80 - P100 ou T4	P100
	Qt. de núcleos	2	2
	RAM	12GB - 26.75GB	13GB
	Espaço de disco	359GB	5GB

Tabela 5 – Comparação entre as plataformas Kaggle e Google Colab/Google Colab Pro.

### 3.4.2 Software

Para a parte da implementação das arquiteturas escolhidas, foram verificadas bibliotecas compatíveis com Python, linguagem escolhida devido à sua facilidade de im-

plementação e suporte para aprendizado de máquina. Algumas das principais bibliotecas utilizadas nessa área são o PyTorch, o Tensorflow e o Keras. Comparações entre as plataformas, linguagens e versões dessas bibliotecas são feitas na Tabela 6.

Software	<b>Keras</b>	<b>TensorFlow</b>	<b>PyTorch</b>
Descrição	API de alto nível para implementação de redes neurais	Plataforma de código aberto para aprendizado de máquina	Plataforma de código aberto para aprendizado de máquina
Plataformas e versões mínimas	macOS 10.12.6 (Sierra), Ubuntu 16.04, Windows 7	macOS 10.12.6 (Sierra), Ubuntu 16.04, Windows 7	macOS 10.10 (Yosemite), Ubuntu 13.04, Windows 7
Python	3.6 – 3.9	3.6 - 3.9	3.x
Outras linguagens	-	Java, C, Go	C++

Tabela 6 – Comparação entre as bibliotecas mais utilizadas para aprendizado profundo de máquina.

O TensorFlow é uma opção muito boa de plataforma de código aberto para aprendizado de máquina, providenciando APIs de alto e baixo nível e possuindo um ecossistema abrangente e flexível de ferramentas, bibliotecas e recursos de comunidade. O Keras, é uma biblioteca de redes neurais de código aberto escrita em Python que oferece APIs de alto nível, possuindo uma melhor interface para o usuário do que as demais. O PyTorch é descrito como uma plataforma de código aberto, constituída por uma biblioteca otimizada para o uso de CPUs e GPUs em aprendizado de máquina. Ele providencia uma API de nível mais baixo, focada no trabalho direto com expressões matriciais. Normalmente, por ser mais utilizado em pesquisas acadêmicas em comparação com os demais, os modelos mais recentes de aprendizado de máquina são encontrados mais facilmente em PyTorch.

O modelo ResNet-RS possui em seu repositório original, sua implementação em TensorFlow 1.x porém, devido às diferenças de compatibilidade entre o TensorFlow 1 e sua versão mais atualizada, o TensorFlow 2, foram verificadas outras opções de implementação. Para isso, o Pytorch se mostrou uma melhor opção, visto que o repositório original da ResNet-RS permitia a importação dos modelos em Pytorch. Outra vantagem desse software foi a possibilidade de aplicação do modelo em nível mais baixo, permitindo uma otimização e melhor compreensão dos passos de treinamento e testes. Essa implementação também possibilitou uma melhor personalização do código, que foi criado de modo que os modelos utilizados pudessem ser alterados com certa facilidade e as classes do banco de dados, referentes à organização de cada pasta, trocadas modificando o mínimo possível de itens<sup>4</sup>.

<sup>4</sup> Repositório com o código de implementação: <[https://github.com/AleMaiaF/ML\\_classification](https://github.com/AleMaiaF/ML_classification)>.

### 3.5 Aplicação e Métricas de desempenho

Para execução do programa, a resolução das imagens foram reduzidas para 224x224, otimizando o GPU utilizado e o tempo de execução. Para auxiliar com o problema de desbalanceamento do número de imagens para cada classe, foi realizada uma distribuição proporcional das imagens por classe entre os conjuntos de treinamento e de teste e na distribuição pelos *folders*.

Para a validação de cada um dos modelos apresentados, foi utilizado o modelo *k-fold*, com o *k* igual à 10, reservando 52 imagens por lote. Devido à proporção da quantidade de lotes de treino e validação, para cada época, a cada 8 lotes de treinamento era utilizado 1 lote de validação para se obter os dados de desempenho da máquina. Tanto a classificação das imagens em famílias quanto a classificação binária em acumuladora ou excludente foram realizadas com a máquina treinada utilizando a função de perda de Entropia Cruzada seguida da função de treinamento *backpropagation* e da função de descida do gradiente, respectivamente. Foi também aplicado o valor de *dropout* de 0,25, sugerido pelo artigo original das ResNets-RS (BELLO et al., 2021).

O número de épocas utilizado é um fator que afeta diretamente o tempo de execução. A quantidade de épocas necessária para se obter os resultados foi determinada de acordo com o tipo de modelo implementado: para os modelos sem pré-treino, foram necessárias 200 épocas de treino para se atingir resultados gráficos e de classificação satisfatórios enquanto para os modelos pré-treinados apenas 20 épocas se fizeram necessárias.

Após as etapas de treino e validação, ao final de cada *fold*, foram feitos os testes da máquina com as imagens do conjunto de testes, obtendo assim a avaliação da máquina treinada. Com isso foi gerada a matriz de confusão referente aos testes e o gráfico com as curvas de aprendizado de treino e validação. Foram também calculados a acurácia, para a classificação geral e dos testes, e a precisão, sensibilidade e *f1-score* para cada classe. Para os casos de classificação binária, acumuladoras (A) ou excludentes (E), foi gerado um gráfico mostrando a curva de aprendizado de acurácia para cada classe durante o processo de validação, mostrando a evolução do treinamento. Após a conclusão de todos os *folders*, foram calculadas as acurácias gerais e por classe e as métricas de precisão, sensibilidade e *f1-score* para cada classe com base nos resultados gerais obtidos em todos os testes, com os desvios padrão referentes aos resultados obtidos dos *folders*. Este processo foi aplicado para os três modelos escolhidos (ResNet-Rs-50,101 e 152) pré-treinados e sem pré-treino, para a implementação os dois tipos de classificação utilizados neste trabalho.

## 4 Resultados

Nesta seção são apresentados os resultados obtidos para cada modelo em cada tipo de classificação aplicada. Vale ressaltar que, devido à proporção de quantidade de lotes de treino e de validação mencionada na seção 3.5, os dados do eixo x (iterações) dos gráficos das curvas de aprendizado apresentam um valor duas vezes maior do que o número de épocas. O código da aplicação utilizada neste relatório pode ser encontrado em repositório público: <[https://github.com/AleMaiaF/ML\\_classification](https://github.com/AleMaiaF/ML_classification)>.

### 4.1 Acumuladoras e Excludentes

Os resultados referentes à classificação das imagens das plantas em acumuladora ou excludente são apresentados a seguir.

#### 4.1.1 Sem pré-treino

Para os modelos sem pré-treino, a arquitetura da ResNet-RS é importada com as devidas configurações, sendo treinada sem a adição de parâmetros iniciais pré-treinados, o que justifica a necessidade da utilização de mais épocas para treino dos parâmetros. Por esse motivo os modelos sem pré-treino utilizaram 200 épocas para concluir suas etapas de treino, validação e teste para cada um dos 10 *folds*.

##### 4.1.1.1 ResNet-RS-50

Para a classificação binária em acumuladoras e excludentes, o modelo ResNet-RS-50, sem pré-treino, rodou com GPU em aproximadamente 3 horas e 50 minutos. Os gráficos de curva de aprendizado de acurácia e de perda podem ser verificados respectivamente nas Figuras 24 e 25. A matriz de confusão gerada com os dados dos testes de todos os *folds* é apresentada na Figura 26. Os resultados de acurácias e desvios padrão podem ser analisados na Tabela 7. Na Tabela 8 são mostradas as métricas gerais de precisão, sensibilidade e *f1-score* para cada classe com seus respectivos desvios padrão.

Visualizando as curvas de aprendizado da Figura 24, mesmo apresentando ruído é possível verificar que existe uma diferença na curva de aprendizado de acurácia durante o treino das classes. A classe E, encontrada em maior quantidade, constituindo aproximadamente 65% do total de imagens do banco, possui uma curva de acurácia acima da classe A, que constitui aproximadamente 35% do bando de imagens. Isso pode indicar uma maior dificuldade de treino no aprendizado das características específicas da classe A quando comparado com a classe E.

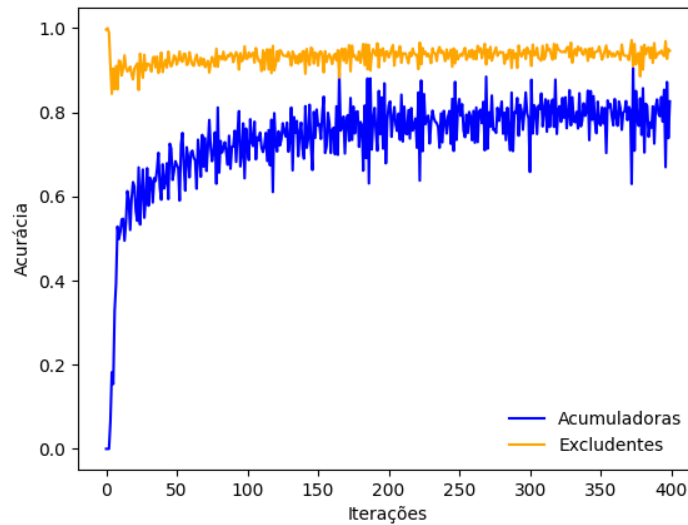


Figura 24 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-50.

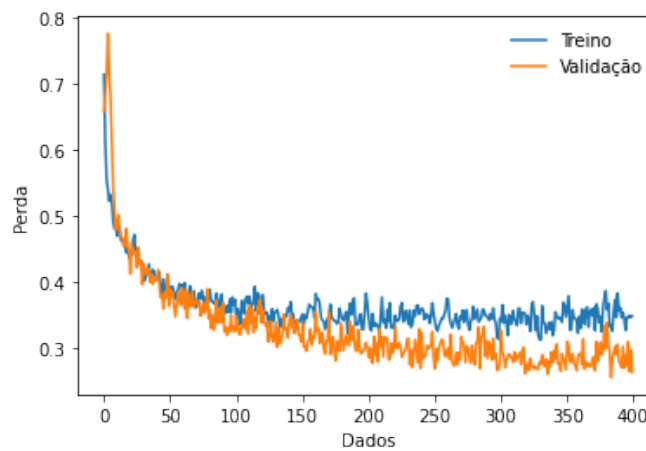


Figura 25 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-50.

Ao analisar as curvas de perda da Figura 25, percebe-se que a curva de validação está abaixo da curva de treino. Isso pode ocorrer quando um banco de dados desbalanceado possui proporções muito discrepantes para a quantidade de itens por classe ou utiliza lotes mal distribuídos para o treino, o que resulta na máquina treinando muito para uma classe e pouco para outra. Poderia ser uma possibilidade, visto que este gráfico é a média dos valores dos gráficos de todos os *folds* e que as imagens foram distribuídas de forma aleatória para os lotes de treino e validação.



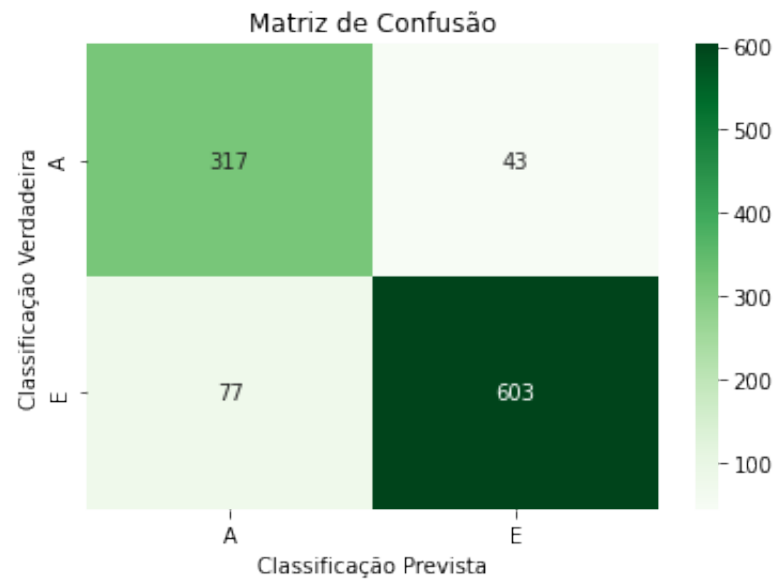


Figura 26 – Matriz de confusão gerada a partir dos dados de teste do modelo ResNet-RS-50 para todos os *folders*.

ResNet-RS-50 - Acurácias		
Classe	Acurácia	Desvio Padrão
A	88.06%	$\pm 4.31\%$
E	88.68%	$\pm 7.15\%$
<b>Geral</b>	<b>88.46 %</b>	<b><math>\pm 4.15\%</math></b>

Tabela 7 – Resultados gerais de acurácia a ResNet-RS-50, com seus respectivos desvios padrão em relação aos valores obtidos em todos os *folders*.

Métricas de Desempenho e Desvios Padrão (DP)						
Classe	Precisão		Sensibilidade		<i>f1-Score</i>	
A	80.46%	$\pm 7.83\%$	88.06%	$\pm 4.31\%$	84.08	$\pm 4.45\%$
E	93.34%	$\pm 2.22\%$	88.68%	$\pm 7.15\%$	90.95	$\pm 3.88\%$

Tabela 8 – Resultados do modelo ResNet-RS-50, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os *folders*.

#### 4.1.1.2 ResNet-RS-101

O modelo ResNet-RS-101 rodou com GPU em aproximadamente 5 horas e 36 minutos. Os gráficos de curva de aprendizado de acurácia e de perda são apresentados respectivamente nas Figuras 27 e 28. A matriz de confusão gerada com os dados dos testes de todos os *folders* é mostrada na Figura 29. As acurácias gerais total e por classe juntamente com seus respectivos desvios padrão *folders* são mostrados na Tabela 9 e as

métricas de desempenho gerais por classe são apresentados na Tabela 10.

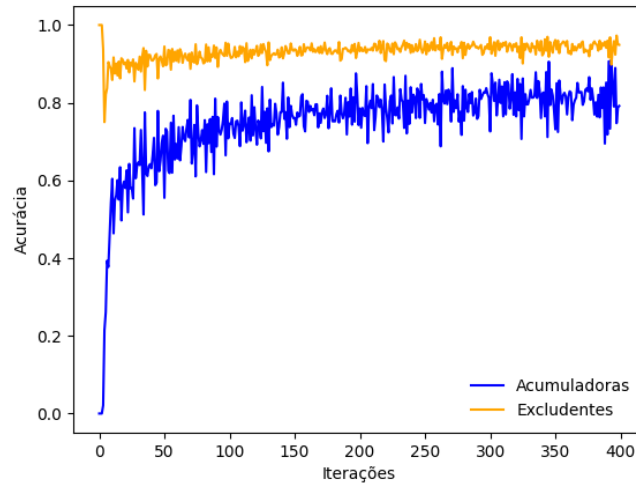


Figura 27 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-101.

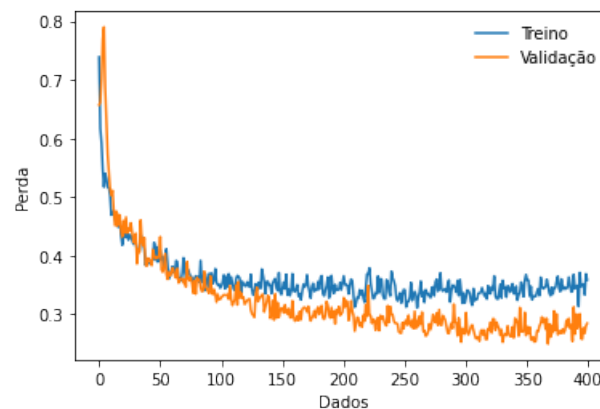


Figura 28 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-101.

As curvas de aprendizado da Figura 27, apresentaram formato semelhante às da ResNet-RS-50, com a classe E possuindo uma curva de acurácia acima da curva da classe A. As curvas de perda da Figura 28, apesar de apresentarem uma maior discrepância ainda possuem comportamento semelhante às curvas de perda da ResNet-RS-50. O comportamento de ambas as curvas pode ocorrer devido à discrepância da quantidade de itens de cada classe no banco de dados, o que dificulta o treino da que existe em menor quantidade enquanto facilita o treino da que existe em abundância.

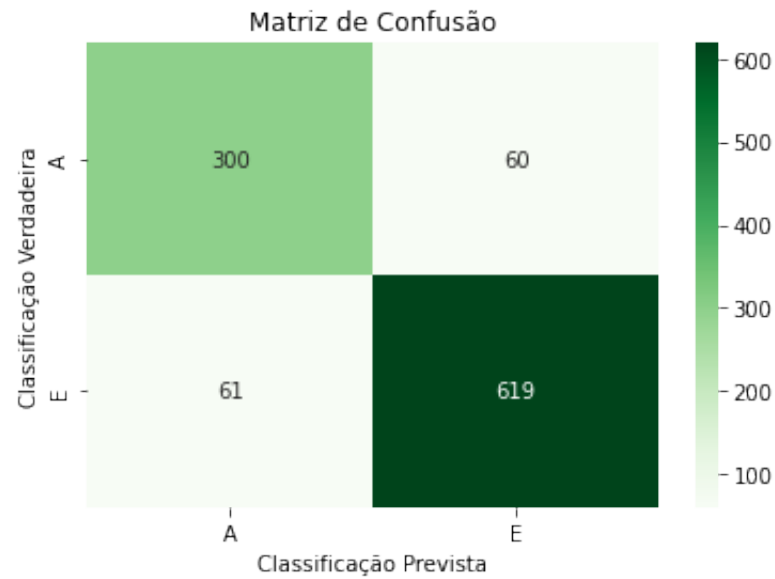


Figura 29 – Matriz de confusão gerada a partir dos dados de teste da ResNet-RS-101 para todos os *folds*.

ResNet-RS-101 - Acurácias		
Classe	Acurácia	Desvio Padrão
<b>A</b>	83,33%	$\pm 10,47\%$
<b>E</b>	91,03%	$\pm 5,24\%$
<b>Geral</b>	88,37 %	$\pm 2,67\%$

Tabela 9 – Resultados gerais de acurácia a ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores obtidos em todos os *folds*.

Métricas de Desempenho e Desvios Padrão (DP)			
Classe	Precisão	Sensibilidade	<i>f1-Score</i>
<b>A</b>	83,10%   $\pm 7,20\%$	83,33%   $\pm 10,47\%$	83,22   $\pm 4,62\%$
<b>E</b>	91,16%   $\pm 4,56\%$	91,03%   $\pm 5,24\%$	91,10   $\pm 2,05\%$

Tabela 10 – Resultados do modelo ResNet-RS-101, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os *folds*.

#### 4.1.1.3 ResNet-RS-152

O modelo ResNet-RS-152 terminou de rodar com GPU em aproximadamente 7 horas e 37 minutos. Os gráficos apresentando as curvas de aprendizado de acurácia e de perda podem ser analisados respectivamente nas Figuras 30 e 31. A matriz de confusão gerada a partir dos dados dos testes de todos os *folders* pode ser verificada na Figura 32. As acurácias gerais e desvios padrão dos resultados dos *folders* são apresentados na Tabela 11 e os resultados gerais das métricas de desempenho para cada classe são mostrados na Tabela 12.

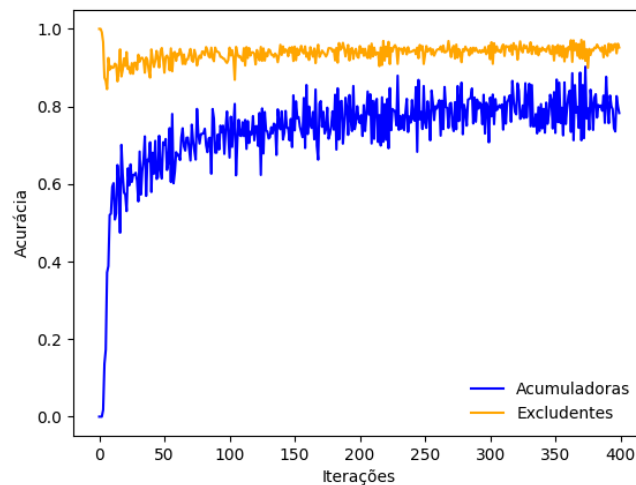


Figura 30 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-152.

As curvas de aprendizado da Figura 30, apresentaram formato semelhante às da ResNet-RS-50 e da ResNet-RS-101, com a classe E possuindo uma curva de acurácia acima da curva da classe A. Podendo indicar uma maior dificuldade de treino no aprendizado das características específicas da classe A quando em comparação com a classe E.

As curvas de perda da Figura 31, apesar de apresentarem uma discrepância ainda maior, elas possuem comportamento semelhante às curvas de perda verificadas nas ResNet-RS-101 e ResNet-RS-50. Apesar do aumento, isso pode ocorrer devido à discrepância da quantidade de itens de cada classe no banco de dados ou nos lotes de imagens de treino e teste, o que dificulta o treino da que existe em menor quantidade enquanto facilita o treino da que existe em abundância. Outra possibilidade é de que devido à aplicação do *dropout* durante o treino, que não é aplicado na validação, os valores de perda de validação tenham ficado menores do que os de treino.

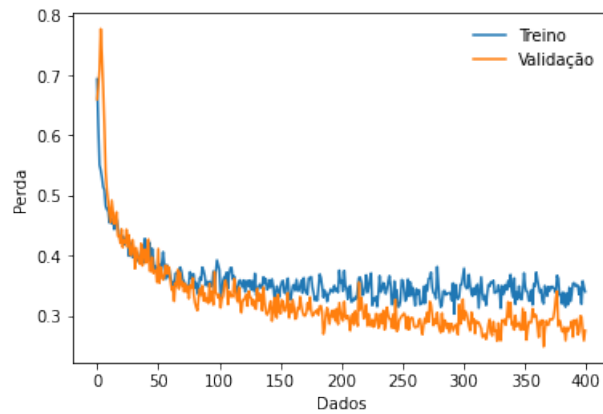


Figura 31 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-152.

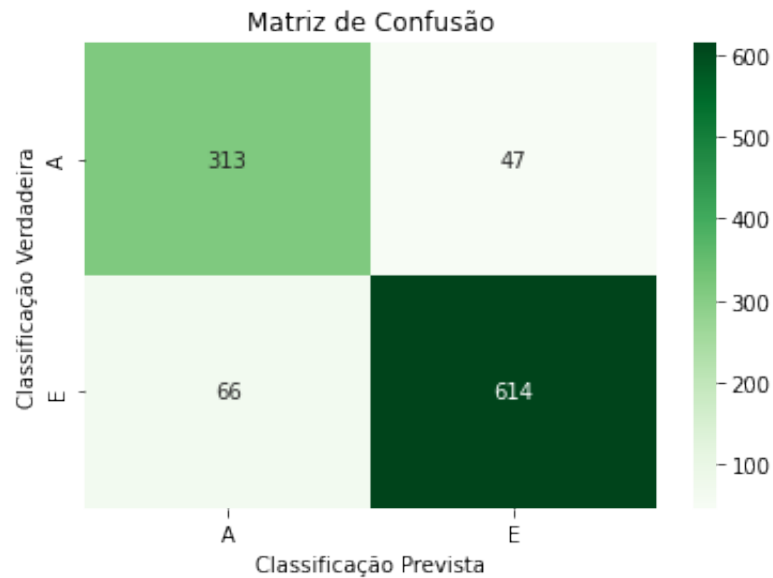


Figura 32 – Matriz de confusão gerada a partir dos dados de teste do modelo ResNet-RS-152 para todos os *folders*.

ResNet-RS-152 - Acurácias		
Classe	Acurácia	Desvio Padrão
A	86,94%	$\pm 9,86\%$
E	90,29%	$\pm 5,55\%$
<b>Geral</b>	<b>89,13 %</b>	<b><math>\pm 1,14\%</math></b>

Tabela 11 – Resultados de acurácia da ResNet-RS-152, com seus respectivos desvios padrão em relação aos valores obtidos em todos os *folders*

Métricas de Desempenho e Desvios Padrão (DP)			
Classe	Precisão	Sensibilidade	<i>f1-Score</i>
<b>A</b>	82,59%   $\pm 8,06\%$	86,94%   $\pm 9,86\%$	84,71   $\pm 2,15\%$
<b>E</b>	92,89%   $\pm 4,30\%$	90,29%   $\pm 5,55\%$	91,57   $\pm 1,09\%$

Tabela 12 – Resultados do modelo ResNet-RS-152, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os *folders*.

## 4.1.2 Pré-treinadas

Para os modelos pré-treinados, foram utilizados os parâmetros pré-treinados retirados do repositório oficial da ResNet-RS. Esses modelos utilizaram 20 épocas para concluir suas etapas de treino, validação e teste para cada um dos 10 *folders*.

### 4.1.2.1 ResNet-RS-50

O modelo ResNet-RS-50 com parâmetros pré-treinados rodou em aproximadamente 22 minutos no GPU do Kaggle. Nas Figuras 33 e 34 são apresentadas, respectivamente, as curvas de aprendizado de acurácia e perda. Na Figura 35 é mostrada a matriz de confusão com os resultados dos testes de todos os *folders*. Nas Tabelas 13 e 14 são mostradas respectivamente as acurácias geral e por classe e as métricas de desempenho por classe todas com seus respectivos desvios padrão para seus valores em todos os *folders*.

Visualizando as curvas de acurácia das classes A e E da Figura 33, é possível notar que elas se mantêm dentro de uma determinada faixa de valores aproximadamente entre 0,9 e 1, apesar da classe A apresentar um pouco mais de ruído do que a classe E. Isso mostra que as características de ambas as classes são aprendidas em graus mais próximos de dificuldade quando em comparação com os gráficos dos modelos sem pré-treino. Com isso, ao se analisar as curvas de perda da Figura 34 e os resultados de matriz de confusão da figura 35, percebe-se que a curva de validação está apenas um pouco abaixo da curva de treino, o que pode ser justificado por essa diferença de dificuldade de treinamento entre as classes.

ResNet-RS-50 pré-treinada - Acurácias		
Classe	Acurácia	Desvio Padrão
<b>A</b>	94,44%	$\pm 3,73\%$
<b>E</b>	98,82%	$\pm 0,59\%$
<b>Geral</b>	97,31%	$\pm 0,94\%$

Tabela 13 – Resultados de acurácia da ResNet-RS-50 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os *folders*.

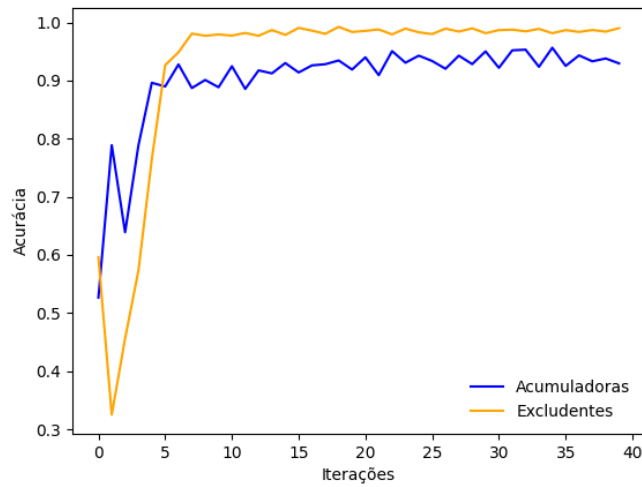


Figura 33 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-50 com parâmetros pré-treinados.

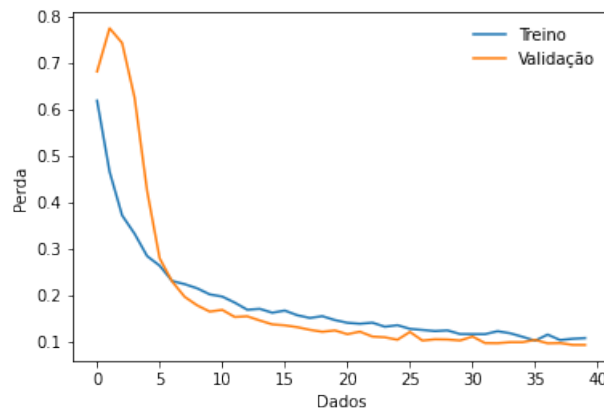


Figura 34 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-50.

Métricas de Desempenho e Desvios Padrão (DP)			
Classe	Precisão	Sensibilidade	<i>f1-Score</i>
<b>A</b>	97,70%   $\pm 1,23\%$	94,44%   $\pm 3,73\%$	96,05%   $\pm 1,52\%$
<b>E</b>	97,11%   $\pm 1,81\%$	98,82%   $\pm 0,59\%$	97,96%   $\pm 0,68\%$

Tabela 14 – Resultados específicos do modelo ResNet-RS-50 pré-treinado, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação às métricas obtidas em todos os *folders*.

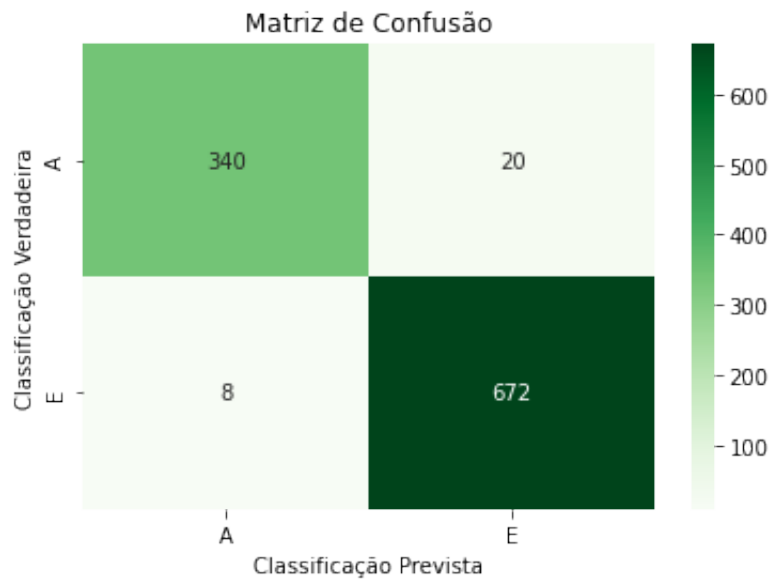


Figura 35 – Matriz de confusão gerada a partir resultados dos testes do modelo pré-treinado ResNet-RS-50, para todos os  *folds*.

#### 4.1.2.2 ResNet-RS-101

O modelo pré-treinado ResNet-RS-101 terminou de rodar no GPU do Kaggle em aproximadamente 35 minutos. As curvas gerais de aprendizado de acurácia e de perda resultantes podem ser verificadas respectivamente nas Figuras 36 e 37. A matriz de confusão gerada a partir dos dados resultantes dos testes de todos os  *folds* pode ser analisada na Figura 38. Na Tabela 15 são apresentados os dados de acurácia com seus respectivos desvios padrão e na Tabela 16 são apresentadas as métricas de desempenho por classe com seus respectivos desvios padrão.

Analisando as curvas de aprendizado da Figura 36 percebe-se uma certa linearidade da curva da classe E em comparação com a curva mais ruidosa da classe A. Isso ocorre pois o gráfico é gerado com os valores médios das curvas obtidas para cada  *fold*. É também observada uma maior perda de treino na Figura 37, podendo indicar uma possível discrepância na dificuldade de treino entre as classes para esse modelo.

ResNet-RS-101 pré-treinada - Acurácias		
Classe	Acurácia	Desvio Padrão
A	98,89%	$\pm 1,84\%$
E	96,76%	$\pm 1,95\%$
<b>Geral</b>	<b>97,5%</b>	<b><math>\pm 1,15\%</math></b>

Tabela 15 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-101 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os  *folds*



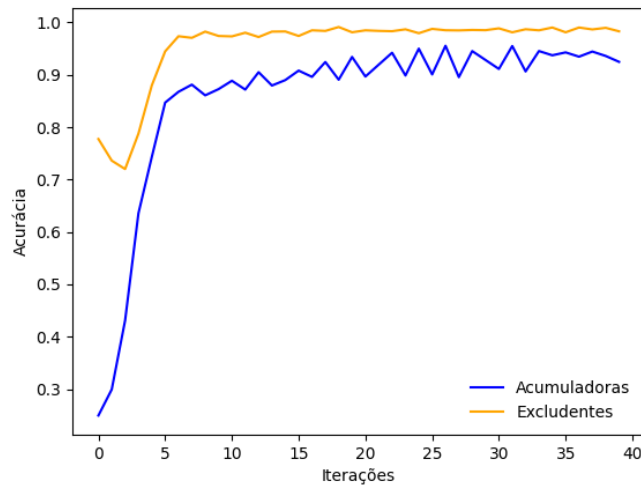


Figura 36 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-101 com parâmetros pré-treinados.

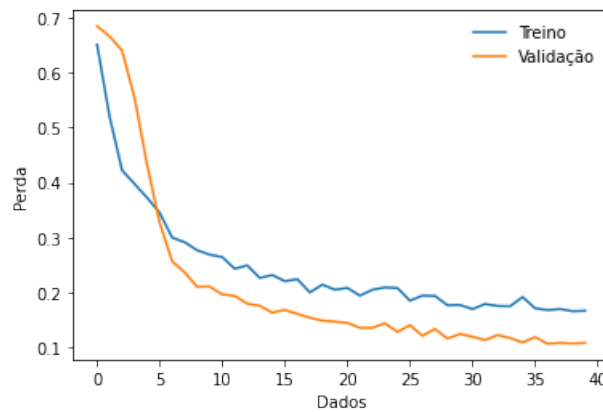


Figura 37 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-101.

Métricas de Desempenho e Desvios Padrão (DP)					
Classe	Precisão		Sensibilidade		<i>f1-Score</i>
<b>A</b>	94,18%	±3,21%	98,89%	1,84±%	96,48%   ±1,57%
<b>E</b>	99,40%	±0,97%	96,76%	±1,95%	98,06%   ±0,91%

Tabela 16 – Resultados do modelo ResNet-RS-101 pré-treinado, mostrando as métricas de desempenho por classe e respectivos desvios padrão em relação aos valores das métricas obtidas em todos os *folders*.

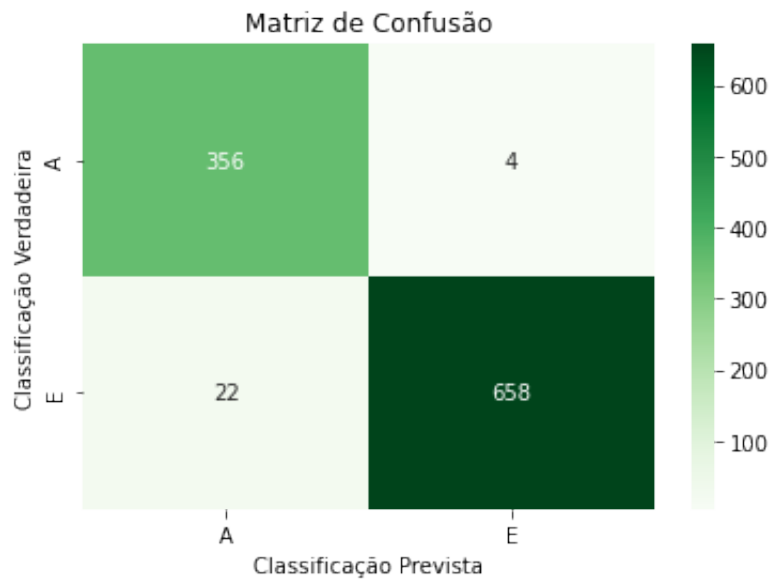


Figura 38 – Matriz de confusão gerada a partir dos dados gerais de teste do modelo pré-treinado ResNet-101 para todos os *folders*.

#### 4.1.2.3 ResNet-RS-152

O modelo pré-treinado ResNet-RS-152 rodou em aproximadamente 48 minutos no GPU do Kaggle. Suas curvas gerais de aprendizado de acurácia e de perda são mostradas nas Figuras 39 e 40 respectivamente. A matriz de confusão gerada utilizando os dados dos testes de todos os *folders* é mostrada na Figura 41. Na Tabela 17 são apresentados os dados de acurácia com seus respectivos desvios padrão e na Tabela 18 são apresentadas as métricas de desempenho por classe com seus respectivos desvios padrão.

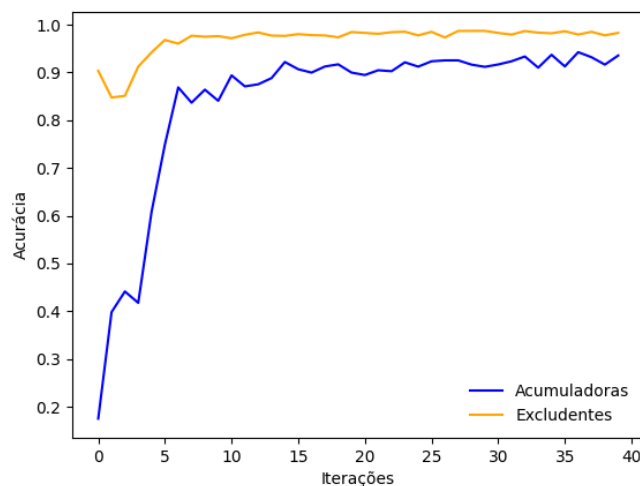


Figura 39 – Gráfico com as curvas de aprendizado de acurácia para as classes A - acumuladora (azul) e E - excludente (laranja) para o treinamento do modelo ResNet-RS-152 com parâmetros pré-treinados.

Na Figura 40 é observada uma maior perda de treino em comparação com a curva de validação, podendo indicar uma possível discrepância na dificuldade de treino entre as classes para esse modelo ResNet-RS em questão.

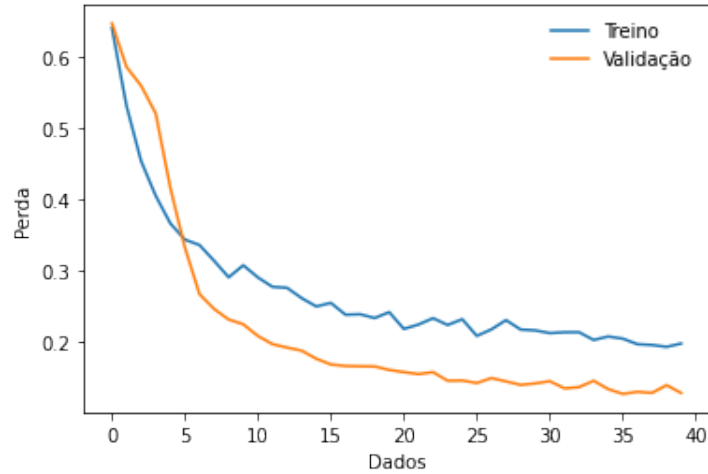


Figura 40 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo pré-treinado ResNet-RS-152.

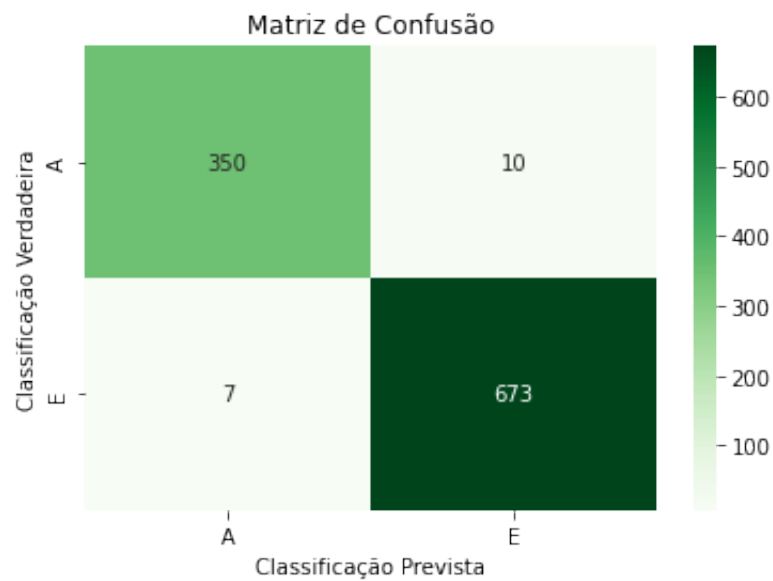


Figura 41 – Matriz de confusão gerada a partir dos dados gerais de teste do modelo pré-treinado ResNet-RS-152 para todos os *folds*.

ResNet-RS-152 pré-treinada - Acurácias		
Classe	Acurácia	Desvio Padrão
<b>A</b>	97,22%	$\pm 2,48\%$
<b>E</b>	98,97%	$\pm 1,15\%$
<b>Geral</b>	98,37 %	$\pm 0,62\%$

Tabela 17 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-152 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os *folders*

Métricas de Desempenho e Desvios Padrão (DP)					
Classe	Precisão		Sensibilidade	<i>f1-Score</i>	
<b>A</b>	98,04%	$\pm 2,07\%$	97,22%	$\pm 2,48\%$	97,63   $\pm 0,92\%$
<b>E</b>	98,54%	$\pm 1,26\%$	98,97%	$\pm 1,15\%$	98,75   $\pm 0,46\%$

Tabela 18 – Resultados do modelo ResNet-RS-152 pré-treinado, com as métricas de desempenho por classe e respectivos desvios padrão em relação a todos os *folders*.

## 4.2 Famílias de Plantas

Os resultados referentes às classificações de acordo com as famílias das plantas das imagens utilizadas no banco são apresentados a seguir.

### 4.2.1 Sem pré-treino

Para os modelos sem pré-treino, a arquitetura da ResNet-RS é importada com as devidas configurações, sendo treinada sem a adição de parâmetros iniciais pré-treinados, o que justifica a necessidade da utilização de mais épocas para treino dos parâmetros. Para isso, para todos os modelos da seção 4.2.1 foram utilizados 200 épocas de treino.

#### 4.2.1.1 ResNet-RS-50

O modelo da ResNet-RS 50 para a classificação das imagens em famílias de plantas rodou em aproximadamente 7 horas e 38 minutos. O gráfico com a curva de perda média geral obtida pode ser verificado na Figura 42. Na Figura 43 pode-se verificar a matriz de confusão gerada a partir de todos os dados de teste de todos os *folders*. As Tabelas 19 e 20 contém respectivamente os resultados gerais de acurácia e as métricas de desempenho, com seus correspondentes desvios padrão, de acordo com todos os *folders*.

Na figura 42 é possível notar que o modelo apresenta um certo sobreajuste, mostrando que durante seu treino ele aprendeu características inerentes aos dados de treino

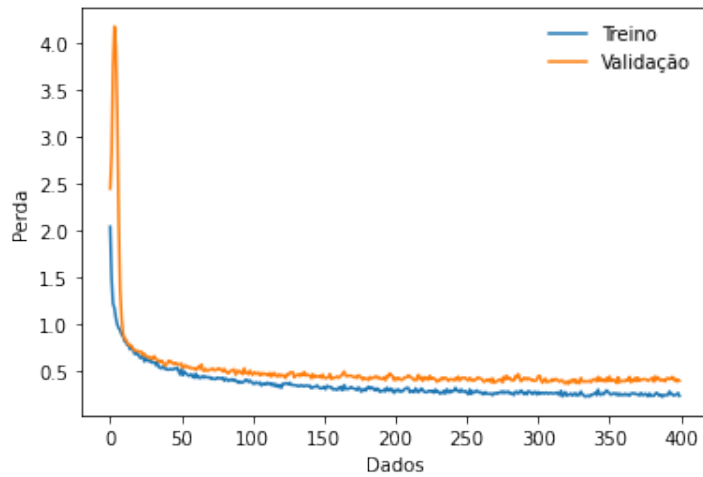


Figura 42 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-50 com as classes de famílias de plantas.

que são irrelevantes para a classificação, podendo obter um desempenho pior do que o esperado em algumas classes enquanto obtém resultados muito bons em outras, com desvios padrão irregulares. Isso pode ser observado em determinadas classes nas Tabelas 19 e 20.

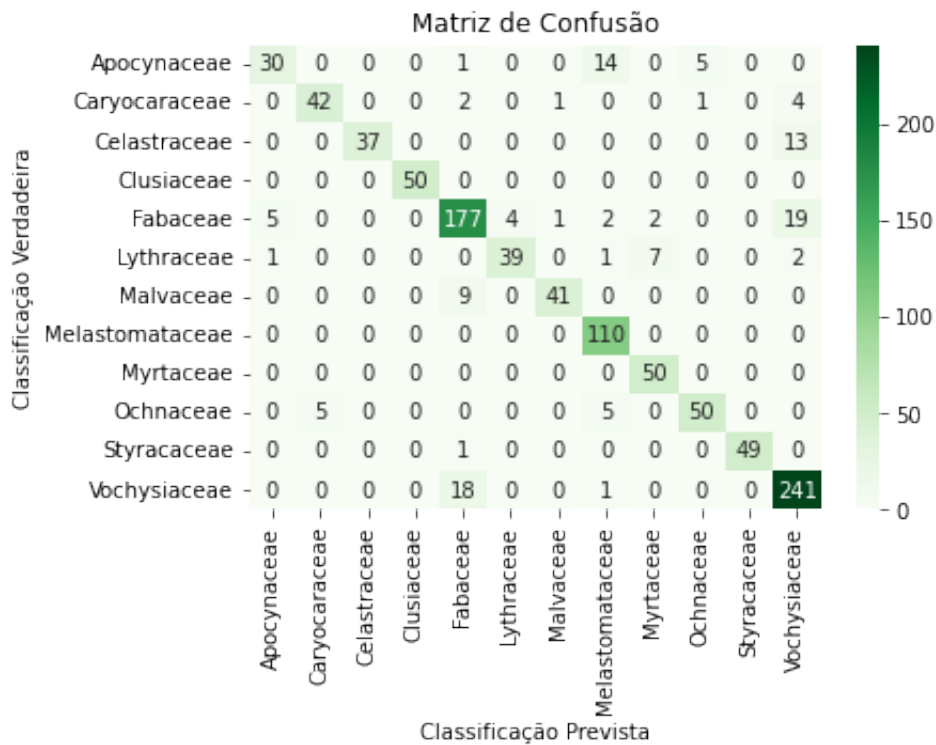


Figura 43 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas em todos os  *folds*  para o modelo ResNet-RS-50 sem pré-treino.

ResNet-RS-50 - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	60,00%	±0,00%
1	Caryocaraceae	84,00%	±8,00%
2	Celastraceae	74,00%	±18,00%
3	Clusiaceae	100,00%	±0,00%
4	Fabaceae	84,29%	±4,29%
5	Lythraceae	78,00%	±10,77%
6	Malvaceae	82,00%	±6,00%
7	Melastomataceae	100,00%	±0,00%
8	Myrtaceae	100,00%	±0,00%
9	Ochnaceae	83,33%	±0,00%
10	Styracaceae	98,00%	±6,00%
11	Vochysiaceae	92,69%	±2,69%
	<b>Geral</b>	88,08 %	±1,23%

Tabela 19 – Resultados de acurácia para a ResNet-RS-50, com seus respectivos desvios padrão em relação às acurácias de todos os *folds*

Métricas de Desempenho e Desvios Padrão (DP)				
Índice	Classe	Precisão	Sensibilidade	<i>f1-Score</i>
0	Apocynaceae	83,33%   ±14,63%	60,00%   ±0,00%	69,77   ±5,19%
1	Caryocaraceae	89,36%   ±9,40%	84,00%   ±8,00%	86,60   ±4,40%
2	Celastraceae	100,00%   ±0,00%	74,00%   ±18,00%	85,06   ±11,30%
3	Clusiaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00   ±0,00%
4	Fabaceae	85,10%   ±5,85%	84,29%   ±4,29%	84,69   ±3,48%
5	Lythraceae	90,70%   ±9,43%	78,00%   ±10,77%	83,87   ±5,93%
6	Malvaceae	95,35%   ±8,57%	82,00%   ±6,00%	88,17   ±1,67%
7	Melastomataceae	82,71%   ±6,36%	100,00%   ±0,00%	90,53   ±3,92%
8	Myrtaceae	84,75%   ±10,24%	100,00%   ±0,00%	91,74   ±5,90%
9	Ochnaceae	89,29%   ±8,16%	83,33%   ±0,00%	86,21   ±3,71%
10	Styracaceae	100,00%   ±0,00%	98,00%   ±6,00%	98,99   ±3,33%
11	Vochysiaceae	86,38%   ±4,25%	92,69%   ±2,69%	89,42   ±2,21%

Tabela 20 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-50 e seus respectivos desvios padrão.

## 4.2.1.2 ResNet-RS-101

O treino do modelo ResNet-RS 101 para a classificação das imagens em famílias de plantas foi realizado em aproximadamente 3 horas e 41 minutos. O gráfico com a curva de perda média geral obtida pode ser verificado na Figura 44. Na Figura 45 é mostrada a matriz de confusão gerada a partir de todos os dados de teste para todos os *folders*. Nas Tabelas 21 e 22 são apresentados respectivamente os resultados gerais de acurácia e as métricas de desempenho, ambos mostrando valores médios dos resultados obtidos para todos os *folders* e seus respectivos desvios padrão.

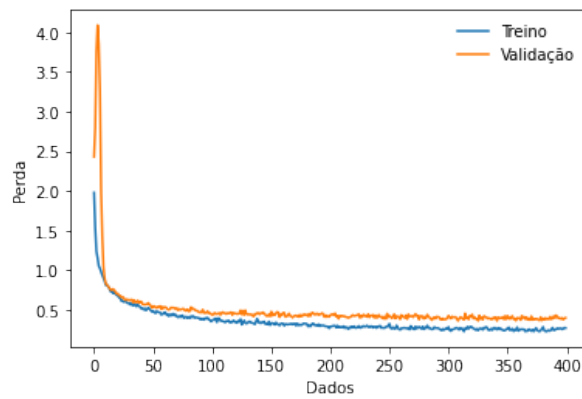


Figura 44 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-101 para as classes de famílias de plantas.

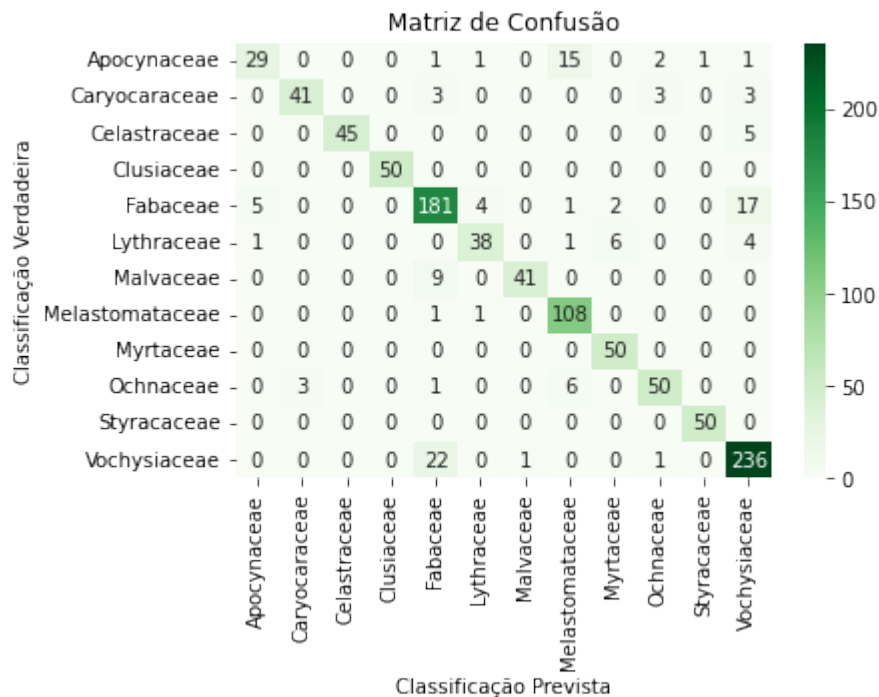


Figura 45 – Matriz de Confusão com os dados de teste do modelo ResNet-RS-101 para a classificação em famílias de plantas para todos os *folders*.

Verificando as curvas de perda da Figura 44, é possível verificar a ocorrência de sobreajuste, o que resulta em menores valores de métricas de desempenho irregulares, como pode ser verificado como consequência nas Tabelas 21 e 22.

ResNet-RS-101 - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	58,00%	±6,00%
1	Caryocaraceae	86,00%	±9,17%
2	Celastraceae	86,00%	±18,00%
3	Clusiaceae	100%	±0%
4	Fabaceae	86,67%	±4,15%
5	Lythraceae	78,00%	±10,77%
6	Malvaceae	82,00%	±6,00%
7	Melastomataceae	99,09%	±2,73%
8	Myrtaceae	100%	±0%
9	Ochnaceae	81,67%	±5,00%
10	Styracaceae	98,00%	±6,00%
11	Vochysiaceae	91,54%	±2,88%
	<b>Geral</b>	88.65 %	±1.04%

Tabela 21 – Resultados das métricas de desempenho para a ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores de todos os *folds*

Métricas de Desempenho e Desvios Padrão (DP)					
Índice	Classe	Precisão	Sensibilidade	<i>f1-Score</i>	
0	Apocynaceae	87,88%   ±14,28%	58,00%   ±6,00%	69,88	±6,61%
1	Caryocaraceae	93,48%   ±8,70%	86,00%   ±9,17%	89,58	±6,36%
2	Celastraceae	100,00%   ±0,00%	86,00%   ±18,00%	92,47	±11,21%
3	Clusiaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
4	Fabaceae	82,35%   ±4,55%	86,67%   ±4,15%	84,45	±3,03%
5	Lythraceae	92,86%   ±8,70%	78,00%   ±10,77%	84,78	±5,99%
6	Malvaceae	97,62%   ±5,00%	82,00%   ±6,00%	89,13	±0,61%
7	Melastomataceae	82,58%   ±6,90%	99,09%   ±2,73%	90,08	±4,58%
8	Myrtaceae	94,34%   ±7,64%	100,00%   ±0,00%	97,09	±4,17%
9	Ochnaceae	89,09%   ±11,70%	81,67%   ±5,00%	85,22	±5,77%
10	Styracaceae	98,00%   ±5,00%	98,00%   ±6,00%	98,00	±4,07%
11	Vochysiaceae	87,18%   ±3,96%	91,54%   ±2,88%	89,31	±2,65%

Tabela 22 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-101 e seus respectivos desvios padrão.

#### 4.2.1.3 ResNet-RS-152

O treino do modelo ResNet-RS 152 para a classificação das imagens em famílias de plantas foi realizado em aproximadamente 7 horas e 38 minutos. O gráfico com a curva



de perda média geral obtida é apresentado na Figura 46. Na Figura 47 pode-se verificar a matriz de confusão gerada a partir de todos os dados de teste de todos os *folders*. Nas Tabelas 23 e 24 são mostrados respectivamente os dados de acurácia e as métricas de desempenho gerais com os respectivos desvios padrão.

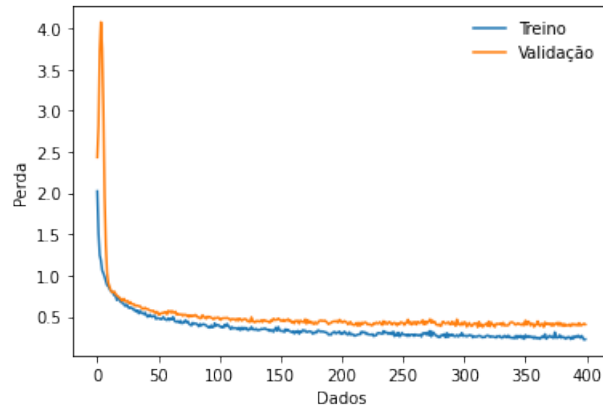


Figura 46 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o treinamento do modelo ResNet-RS-152 para as classes de famílias de plantas.

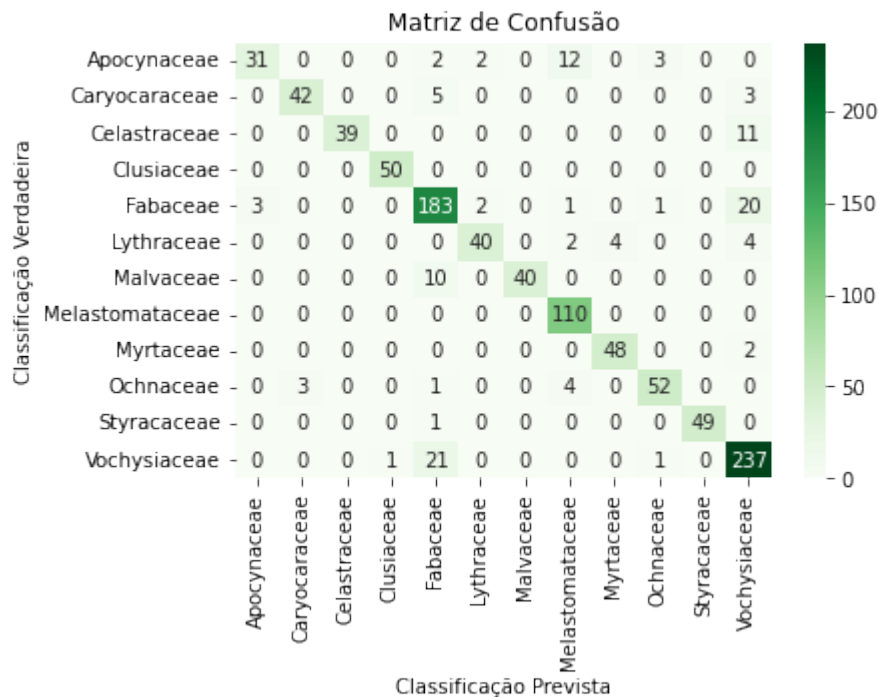


Figura 47 – Matriz de Confusão com os dados de teste para a classificação entre as famílias das plantas de todos os *folders* para o modelo ResNet-RS-152.

Observando as curvas de perda apresentadas na Figura 46, é possível notar a ocorrência de um certo sobreajuste, o que tem como consequência valores de métricas de desempenho mais irregulares, como pode ser verificado nas Tabelas 23 e 24.

ResNet-RS-152 - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	56,00%	±8,00%
1	Caryocaraceae	86,00%	±9,17%
2	Celastraceae	76,00%	±17,44%
3	Clusiaceae	100,00%	±0,00%
4	Fabaceae	89,52%	±6,67%
5	Lythraceae	80,00%	±0,00%
6	Malvaceae	82,00%	±6,00%
7	Melastomataceae	98,18%	±3,64%
8	Myrtaceae	100,00%	±0,00%
9	Ochnaceae	86,67%	±6,67%
10	Styracaceae	98,00%	±6,00%
11	Vochysiaceae	90,00%	±3,53%
	<b>Geral</b>	88,65 %	±1,04%

Tabela 23 – Resultados de acurácia para a ResNet-RS-152, com seus respectivos desvios padrão em relação aos resultados de todos os *folds*

Métricas de Desempenho e Desvios Padrão (DP)				
Índice	Classe	Precisão	Sensibilidade	<i>f1-Score</i>
0	Apocynaceae	82,35%   ±13,15%	56,00%   ±8,00%	66,67   ±7,59%
1	Caryocaraceae	95,56%   ±6,67%	86,00%   ±9,17%	90,53   ±3,30%
2	Celastraceae	100,00%   ±0,00%	76,00%   ±17,44%	86,36   ±10,97%
3	Clusiaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00   ±0,00%
4	Fabaceae	81,03%   ±4,92%	89,52%   ±6,67%	85,07   ±4,16%
5	Lythraceae	93,02%   ±11,08%	80,00%   ±0,00%	86,02   ±5,27%
6	Malvaceae	100,00%   ±0,00%	82,00%   ±6,00%	90,11   ±3,33%
7	Melastomataceae	83,08%   ±5,03%	98,18%   ±3,64%	90,00   ±3,71%
8	Myrtaceae	83,33%   ±12,81%	100,00%   ±0,00%	90,91   ±7,46%
9	Ochnaceae	91,23%   ±9,98%	86,67%   ±6,67%	88,89   ±6,00%
10	Styracaceae	100,00%   ±0,00%	98,00%   ±6,00%	98,99   ±3,33%
11	Vochysiaceae	89,66%   ±5,43%	90,00%   ±3,53%	89,83   ±3,02%

Tabela 24 – Métricas de desempenho por classe do modelo ResNet-RS-152 e seus respectivos desvios padrão.

#### 4.2.2 Pré-treinadas

Esses modelos tiveram seus parâmetros pré-treinados retirados do repositório oficial da ResNet-RS. Para esses modelos pré-treinados, foram utilizados no total 20 épocas para os processos de treino, validação e testes de cada um dos 10 *folds*.

## 4.2.2.1 ResNet-RS-50

O treino do modelo ResNet-RS-50 pré-treinado para a classificação das imagens em famílias de plantas foi realizado em aproximadamente 23 minutos. O gráfico com as curvas de perda gerais obtidas é apresentado na Figura 48. Na Figura 49 é mostrada a matriz de confusão gerada a partir dos dados de teste de todos os *folders*. As Tabelas 25 e 26 contém respectivamente os resultados gerais de acurácia e as métricas de desempenho, como médias gerais obtidas de todos os *folders* e seus respectivos desvios padrão.

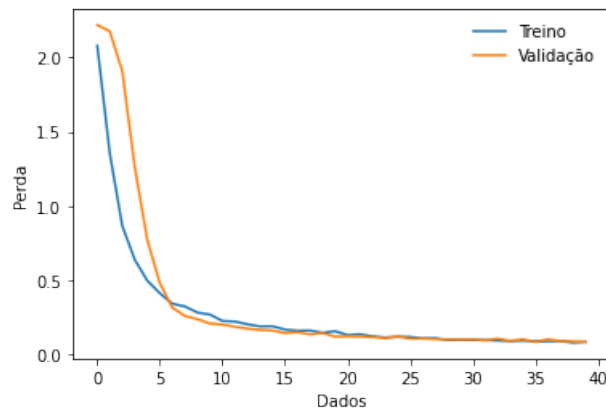


Figura 48 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-50 pré-treinado para as classes de famílias de plantas.

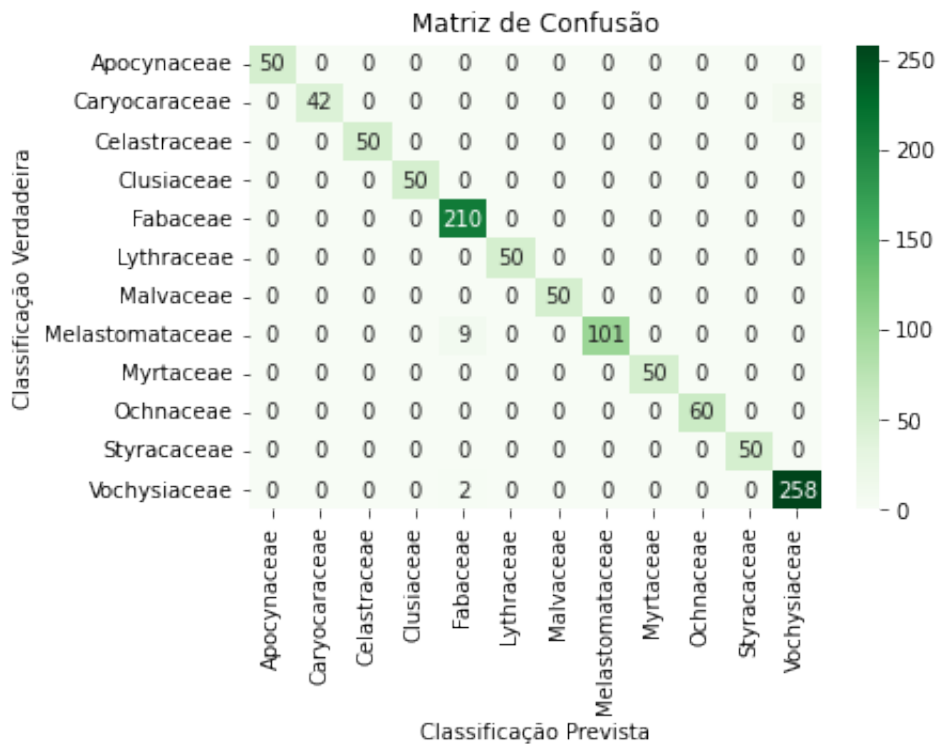


Figura 49 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas de todos os *folders* para o modelo ResNet-RS-50 pré-treinado.

ResNet-RS-50 pré-treinada - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	100,00%	±0,00%
1	Caryocaraceae	84,00%	±8,00%
2	Celastraceae	100,00%	±0,00%
3	Clusiaceae	100,00%	±0,00%
4	Fabaceae	100,00%	±0,00%
5	Lythraceae	100,00%	±0,00%
6	Malvaceae	100,00%	±0,00%
7	Melastomataceae	91,82%	±2,73%
8	Myrtaceae	100,00%	±0,00%
9	Ochnaceae	100,00%	±0,00%
10	Styracaceae	100,00%	±0,00%
11	Vochysiaceae	99,23%	±1,54%
	<b>Geral</b>	98.17 %	±0.52%

Tabela 25 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-50 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os *folders*

Métricas de Desempenho e Desvios Padrão (DP)							
Índice	Classe	Precisão		Sensibilidade		<i>f1-Score</i>	
0	Apocynaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
1	Caryocaraceae	100,00%	±0,00%	84,00%	±8,00%	91,30	±4,44%
2	Celastraceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
3	Clusiaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
4	Fabaceae	95,02%	±2,32%	100,00%	±0,00%	97,45	±1,22%
5	Lythraceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
6	Malvaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
7	Melastomataceae	100,00%	±0,00%	91,82%	±2,73%	95.73	±1,43%
8	Myrtaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
9	Ochnaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
10	Styracaceae	100,00%	±0,00%	100,00%	±0,00%	100,00	±0,00%
11	Vochysiaceae	96,99%	±1,49%	99,23%	±1,54%	98,10	±0,86%

Tabela 26 – Tabela com as métricas de desempenho por classe do modelo pré-treinado ResNet-RS-50 e seus respectivos desvios padrão.

Analisando os dados da matriz de confusão da Figura 49 percebe-se que o maior número de erros de classificação por classe foram das 8 imagens da família Caryocaraceae, classificadas como Vochysiaceae o que resultou em menores sensibilidade e acurácia nas métricas gerais para a classe Caryocaraceae.

## 4.2.2.2 ResNet-RS-101

O modelo ResNet-RS 101 pré-treinado para a classificação das imagens em famílias de plantas rodou no GPU do Kaggle em aproximadamente 34 minutos. O gráfico com as curvas de perda média gerais é mostrado na Figura 50. A matriz de confusão gerada a partir dos dados de teste de todos os *folders* é apresentada na Figura 51. As Tabelas 27 e 28 mostram respectivamente os dados gerais de acurácia e as métricas gerais de desempenho, com seus respectivos desvios padrão.

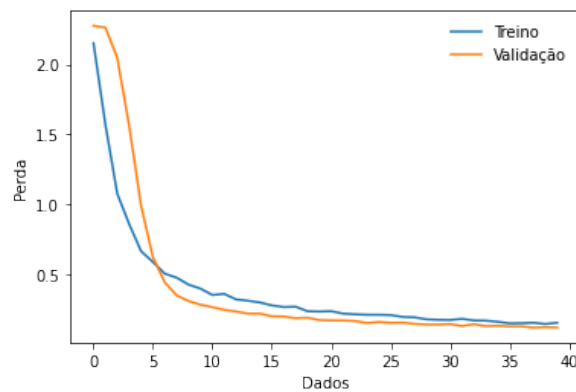


Figura 50 – Gráfico com as curvas de perda de validação (laranja) e treino (azul) para o modelo ResNet-RS-101 pré-treinado para as classes de famílias de plantas.

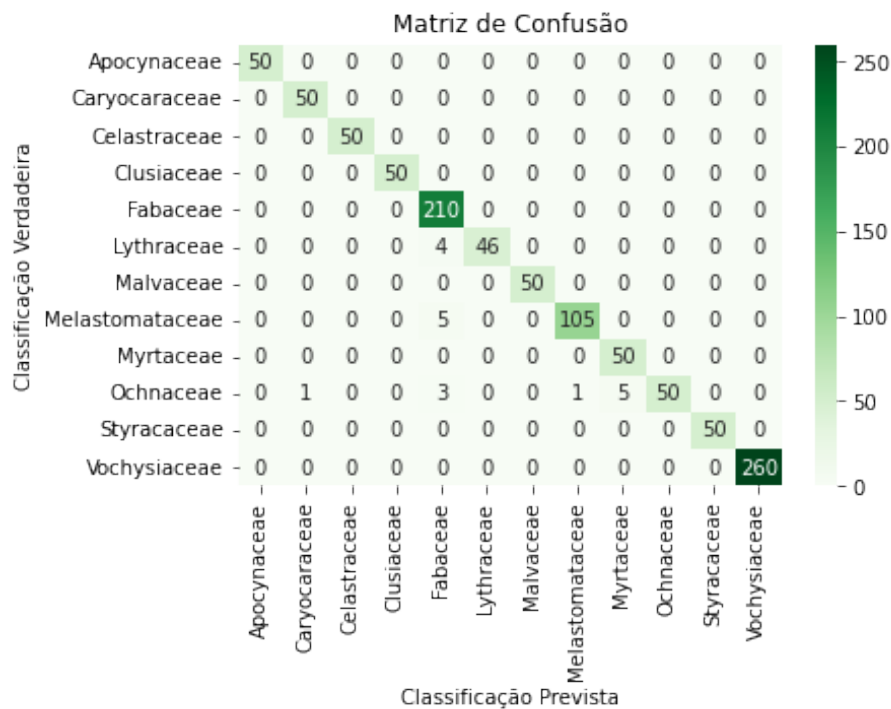


Figura 51 – Matriz de Confusão com os dados de teste para a classificação entre as famílias das plantas do banco de dados de todos os *folders* para o modelo ResNet-RS-101 sem pré-treinado.

ResNet-RS-101 pré-treinada - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	100,00%	±0,00%
1	Caryocaraceae	100,00%	±0,00%
2	Celastraceae	100,00%	±0,00%
3	Clusiaceae	100,00%	±0,00%
4	Fabaceae	100,00%	±0,00%
5	Lythraceae	92,00%	±9,80%
6	Malvaceae	100,00%	±0,00%
7	Melastomataceae	95,45%	±4,55%
8	Myrtaceae	100,00%	±0,00%
9	Ochnaceae	83,33%	±0,00%
10	Styracaceae	100,00%	±0,00%
11	Vochysiaceae	100,00%	±0,00%
	<b>Geral</b>	98,17 %	±0,67%

Tabela 27 – Resultados de acurácia do modelo pré-treinado ResNet-RS-101, com seus respectivos desvios padrão em relação aos valores obtidos em todos os *folds*

No geral, o modelo obteve um bom desempenho. Porém, analisando a matriz de confusão da Figura 51 e a acurácia obtida para a classe da família Ochnaceae com desvio padrão igual à 0 nas Tabelas 27 e 28, pode ser sugerido que, pelo menos para essa classe, a máquina pode estar classificando errado alguma imagem específica do banco ou que, para cada um dos *folds*, ela classificou erroneamente apenas uma imagem.

Métricas de Desempenho e Desvios Padrão (DP)					
Índice	Classe	Precisão	Sensibilidade	<i>f1-Score</i>	
0	Apocynaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
1	Caryocaraceae	98,04%   ±5,00%	100,00%   ±0,00%	99,01	±2,73%
2	Celastraceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
3	Clusiaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
4	Fabaceae	94,59%   ±3,66%	100,00%   ±0,00%	97,22	±1,94%
5	Lythraceae	100,00%   ±0,00%	92,00%   ±9,80%	95,83	±5,44%
6	Malvaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
7	Melastomataceae	99,06%   ±2,73%	95,45%   ±4,55%	97,22	±3,07%
8	Myrtaceae	90,91%   ±8,33%	100,00%   ±0,00%	95,24	±4,55%
9	Ochnaceae	100,00%   ±0,00%	83,33%   ±0,00%	90,91	±0,00%
10	Styracaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%
11	Vochysiaceae	100,00%   ±0,00%	100,00%   ±0,00%	100,00	±0,00%

Tabela 28 – Tabela Com as métricas de desempenho por classe do modelo ResNet-RS-101 pré-treinado e seus respectivos desvios padrão.

## 4.2.2.3 ResNet-RS-152

O treino do modelo da ResNet-RS 152 pré-treinado para a classificação das imagens em famílias de plantas foi realizado em aproximadamente 49 minutos. O gráfico com a curva de perda média geral obtida pode ser verificado na Figura 52. Na Figura 53 pode-se verificar a matriz de confusão gerada a partir de todos os dados de teste de todos os *folders*. As Tabelas 29 e 30 contém respectivamente os resultados gerais de acurácia e as métricas de desempenho obtidos de todos os *folders*, com seus respectivos desvios padrão.

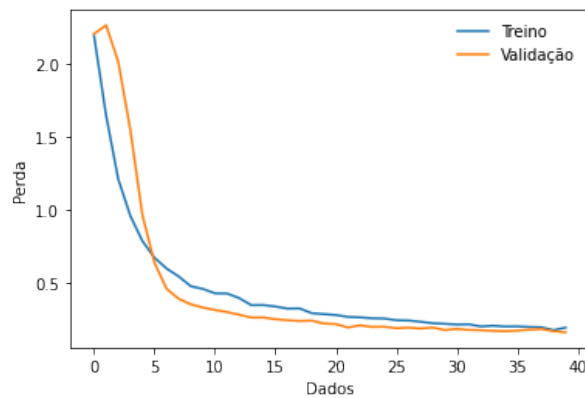


Figura 52 – Gráfico com as curvas de perda dos processos de validação (laranja) e treino (azul) para o modelo ResNet-RS-152 pré-treinado, para as classes de famílias de plantas.

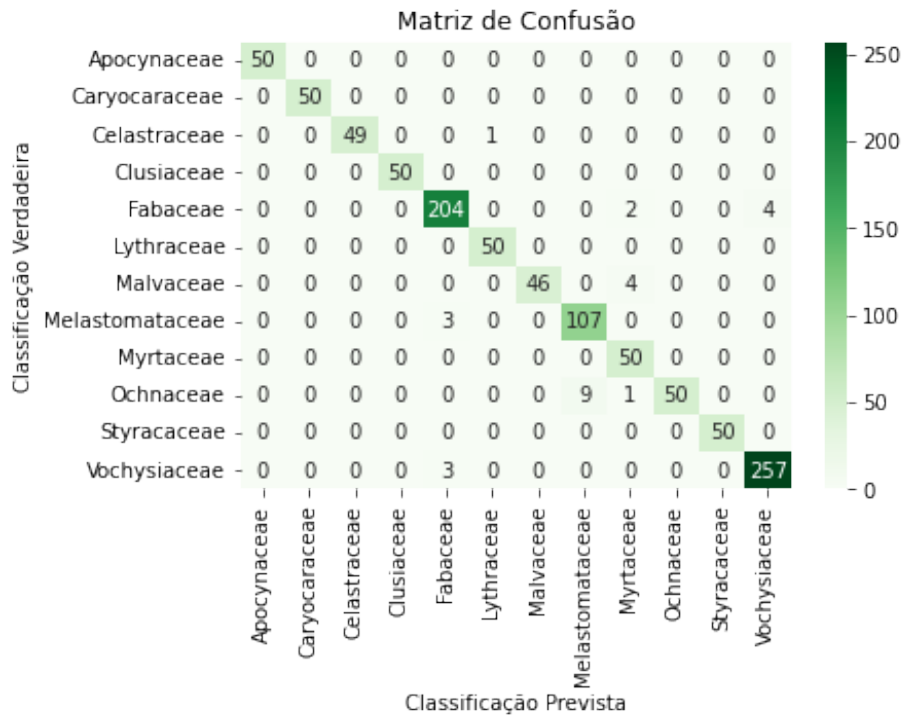


Figura 53 – Matriz de Confusão com os dados de teste para a classificação entre famílias de plantas de todos os *folders* para o modelo ResNet-RS-152 pré-treinado.

ResNet-RS-152 pré-treinada - Acurácias			
Índice	Classe	Acurácia	Desvio Padrão
0	Apocynaceae	100,00%	±0,00%
1	Caryocaraceae	100,00%	±0,00%
2	Celastraceae	98,00%	±6,00%
3	Clusiaceae	100,00%	±0,00%
4	Fabaceae	97,14%	±3,16%
5	Lythraceae	100,00%	±0,00%
6	Malvaceae	92,00%	±9,80%
7	Melastomataceae	97,27%	±4,17%
8	Myrtaceae	100,00%	±0,00%
9	Ochnaceae	83,33%	±0,00%
10	Styracaceae	100,00%	±0,00%
11	Vochysiaceae	98,85%	±1,76%
	<b>Geral</b>	97,40 %	±0,97%

Tabela 29 – Resultados de acurácia das métricas de desempenho para a ResNet-RS-152 pré-treinada, com seus respectivos desvios padrão em relação às acurácias obtidas em todos os *folders*

Métricas de Desempenho e Desvios Padrão (DP)						
Índice	Classe	Precisão		Sensibilidade		<i>f1-Score</i>
0	Apocynaceae	100,00%	±0,00%	100,00%	±0,00%	100,00   ±0,00%
1	Caryocaraceae	100,00%	±0,00%	100,00%	±0,00%	100,00   ±0,00%
2	Celastraceae	100,00%	±0,00%	98,00%	±6,00%	98,99   ±3,33%
3	Clusiaceae	100,00%	±0,00%	100,00%	±0,00%	100,00   ±0,00%
4	Fabaceae	97,14%	±3,03%	97,14%	±3,16%	97,14   ±2,07%
5	Lythraceae	98,04%	±5,00%	100,00%	±0,00%	99,01   ±2,73%
6	Malvaceae	100,00%	±0,00%	92,00%	±9,80%	95,83   ±5,44%
7	Melastomataceae	92,24%	±2,60%	97,27%	±4,17%	94,69   ±2,77%
8	Myrtaceae	87,72%	±11,48%	100,00%	±0,00%	93,46   ±6,60%
9	Ochnaceae	100,00%	±0,00%	83,33%	±0,00%	90,91   ±0,00%
10	Styracaceae	100,00%	±0,00%	100,00%	±0,00%	100,00   ±0,00%
11	Vochysiaceae	98,47%	±1,81%	98,85%	±1,76%	98,66   ±0,88%

Tabela 30 – Tabela com as métricas de desempenho por classe do modelo pré-treinado ResNet-RS-152 e seus respectivos desvios padrão.

Observando os dados das Tabelas 29 e 30, percebe-se que o modelo ResNet-RS-152 obteve resultados gerais bons porém, com desempenhos piores quando comparado com os das ResNet-Rs-101 e ResNet-RS-50 pré-treinadas.



## 4.3 Resultados Gerais e Discussões

Para a melhor análise dos resultados obtidos, foi avaliada qual métrica teria um maior foco para os valores gerais, a acurácia ou a *f1-score*. A *f1-score* é utilizada quando a quantidade de resultados falso negativos e falso positivos é mais importante, por esse motivo a métrica é bastante utilizada na avaliação de imagens médicas. A acurácia é utilizada mais quando a quantidade de positivos verdadeiros e negativos verdadeiros é mais importante. Neste trabalho, para uma avaliação geral dos testes de cada modelo houve um foco maior nos valores de acurácia. Porém, nas avaliações dos resultados dos testes para cada classe foram obtidas as métricas de desempenho de precisão (para avaliar os resultados positivos para a classe em questão), a sensibilidade (para avaliar os resultados negativos para a classe em questão) e da *f1-score* (para avaliar a média harmônica das duas métricas anteriores).

Ao final, foram geradas tabelas com todos os valores gerais de acurácia e desvios padrão obtidos pelos modelos juntamente com a duração do tempo de treino para comparações de desempenho. Como o desvio padrão é tido como um indicador da uniformidade dos dados, e como todas essas métricas gerais obtidas são as médias dos valores obtidos para cada *fold* dos processos de treinamento dos modelos, essa medida ao lado dos valores das métricas mostra o quanto esse valor variou dentre os *folds*, podendo ser utilizada como uma métrica da confiabilidade dos valores obtidos ao se treinar o modelo uma única vez. Importante acrescentar que nas tabelas e figuras apresentadas nessa seção, a sigla “ptr” é utilizada para demarcar os modelos que utilizaram parâmetros pré-treinados.

### 4.3.1 Classificação em Acumuladoras e Excludentes

Na Tabela 31 pode ser verificada uma comparação dos resultados gerais de acurácia de todos os modelos juntamente com seus respectivos desvios padrão e o tempo total em que o modelo em questão demorou para executar, o que inclui os processos de treino, validação e testes para os 10 *folds*. Na Figura 54 foi feita uma tradução dos dados da Tabela 31, com uma conversão dos valores do tempo total de minutos para horas para uma melhor visualização dos dados. Lembrando que foram utilizadas 20 épocas para os modelos pré-treinados e 200 para os não pré-treinados, o que influencia no valor de tempo total dos modelos. Outro fator que influencia na duração do tempo total é o tamanho das arquiteturas, sendo os valores de 50, 101 e 152 referentes à quantidade de camadas que elas utilizam.

Analisando os valores da Tabela 31 e da Figura 54 percebe-se que os modelos mais complexos ResNet-RS-152 obtiveram maiores valores de acurácia e menores valores de desvio padrão comparando com as ResNet-RS-101 e ResNet-RS-50 tanto para os modelos com parâmetros pré-treinados quanto para os sem esses parâmetros. Uma ocorrência

curiosa, porém foi a maior acurácia média obtida pelo modelo ResNet-RS-50 sem pré-treino, que foi mais alto do que a do modelo 101. Seu desvio padrão porém, foi mais alto, indicando que caso o treinamento fosse refeito, essas acurácias poderiam obter uma maior alteração. Esta ocorrência é de certa forma invertida nos modelos pré-treinados, onde a ResNet-RS-50 obteve um menor desvio padrão quando comparado com o modelo 101 apesar de ter tido uma acurácia menor.

Classificação A/E - Acurácias Gerais e Desvios Padrão			
ResNet-RS	Acurácia	Desvio Padrão	Tempo Total (min)
<b>50</b>	88,46 %	$\pm 4,15\%$	220 min
<b>101</b>	88,37 %	$\pm 2,67\%$	335 min
<b>152</b>	<b>89,13 %</b>	<b><math>\pm 1,14\%</math></b>	457 min
<b>50 ptr</b>	97,31 %	$\pm 0,94\%$	22 min
<b>101 ptr</b>	97,5 %	$\pm 1,15\%$	35 min
<b>152 ptr</b>	<b>98,37 %</b>	<b><math>\pm 0,62\%</math></b>	48 min

Tabela 31 – Dados resultantes de acurácias gerais, desvios padrão e tempo total para o treino, validação e testes para os 10 *folds* de todos os modelos ResNet-RS utilizados para a classificação das imagens das plantas em acumuladoras ou excludentes.

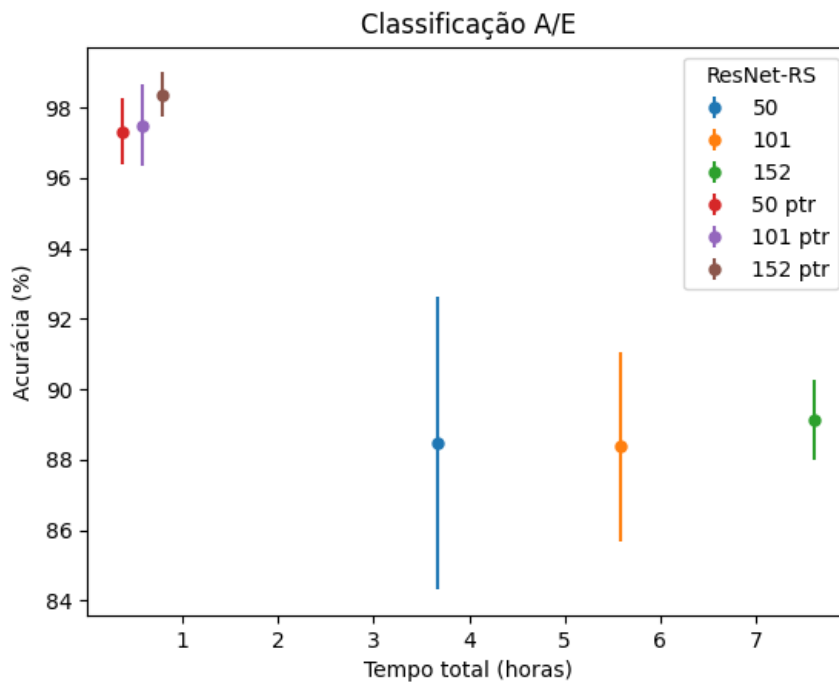


Figura 54 – Tradução dos dados da Tabela 31, comparando as acurácias obtidas com tempo que cada modelo demorou para terminar de rodar. As barras coloridas representam os desvios padrão de cada modelo ResNet-RS.

Na Tabela 32 são apresentadas as métricas de desempenho obtidas com seus respectivos desvios padrão para cada classe em todos os modelos. Os maiores valores de *f1-score* e os respectivos menores desvios padrão para cada classe dentre os modelos sem pré-treino e pré-treinados são destacados dentre os resultados. Para uma melhor comparação e visualização dos resultados, os dados da Tabela 32 foram traduzidos para as Figuras 55, 56 e 57, realizando comparações entre os resultados obtidos de precisão, sensibilidade e *f1-score* respectivamente.

Classificação A/E - Métricas de Desempenho e Desvios Padrão							
ResNet-RS	Classe	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
50	A	80,46 %	±7,83%	88,06 %	±4,31%	84,08 %	±4,45%
	E	93,34 %	±2,22%	88,68 %	±7,15%	90,95 %	±3,88%
101	A	83,1%	±7,2%	83,33%	±10,47%	83,22%	±4,62%
	E	91,16 %	±4,56%	91,03 %	±5,24%	91,1 %	±2,05%
152	A	82,59 %	±8,06%	86,94 %	±9,86%	<b>84,71 %</b>	<b>±2,15%</b>
	E	92,89 %	±4,3%	90,29 %	±5,55%	<b>91,57 %</b>	<b>±1,09%</b>
50 ptr	A	97,7 %	±1,12%	94,44 %	±3,73%	96,05 %	±1,52%
	E	97,11 %	±1,81%	98,82 %	±1,59%	97,96 %	±0,68%
101 ptr	A	94,18 %	±3,21%	98,89 %	±1,84%	96,48 %	±1,57%
	E	99,4 %	±0,97%	96,76 %	±1,95%	98,06 %	±0,91%
152 ptr	A	98,04 %	±2,07%	97,22 %	±2,48%	<b>97,63 %</b>	<b>±0,92%</b>
	E	98,54 %	±1,26%	98,97 %	±1,15%	<b>98,75 %</b>	<b>±0,46%</b>

Tabela 32 – Dados resultantes das métricas de desempenho por classe e desvios padrão para todos os modelos ResNet-RS utilizados para a classificação das imagens das plantas em acumuladoras ou excludentes. Valores obtidos como médias de seus respectivos valores de todos os *folds*.

Analisando os resultados mostrados na Tabela 32 e das Figuras 55, 56 e 57 é possível perceber que, de maneira geral, os modelos pré-treinados apresentaram melhores resultados de classificação do que os modelos sem pré-treino. Como esses modelos pré-treinados tiveram seus parâmetros treinados com um banco de imagens muito grande e diverso como o da ImageNet, tem-se que suas camadas, principalmente as iniciais, que normalmente são responsáveis por reconhecer os elementos mais básicos de uma figura, obtiveram um treino mais eficiente para o reconhecimento de elementos básicos mais gerais, necessitando apenas adaptar suas camadas mais profundas para o reconhecimento e separação das características pertencentes a cada classe.

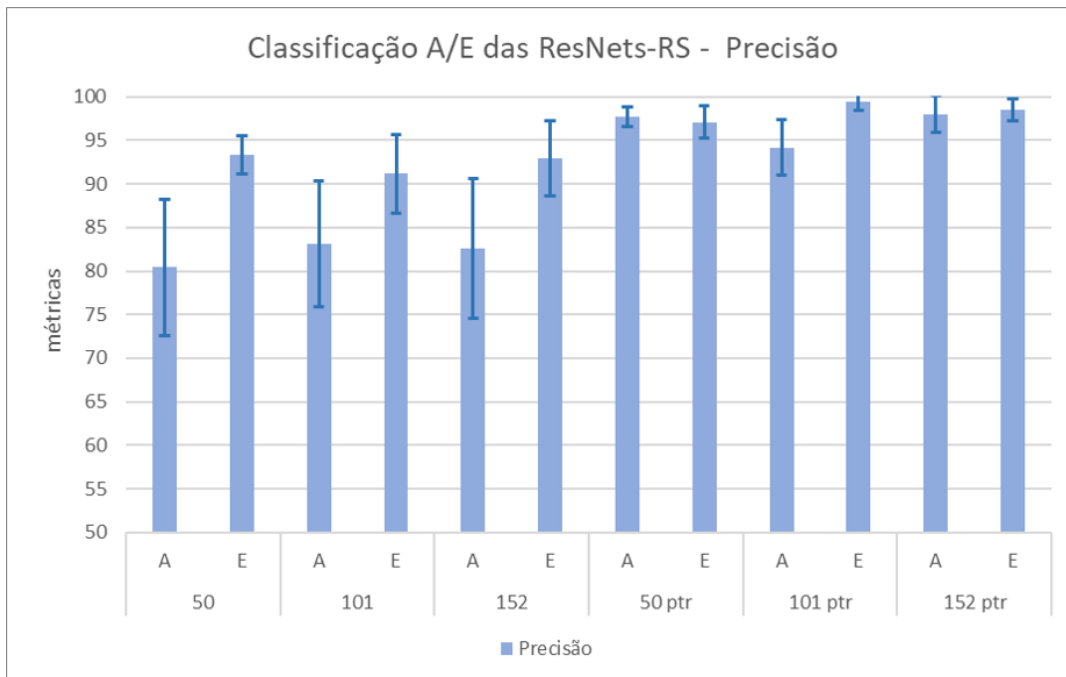


Figura 55 – Comparação dos valores de precisão, em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32.

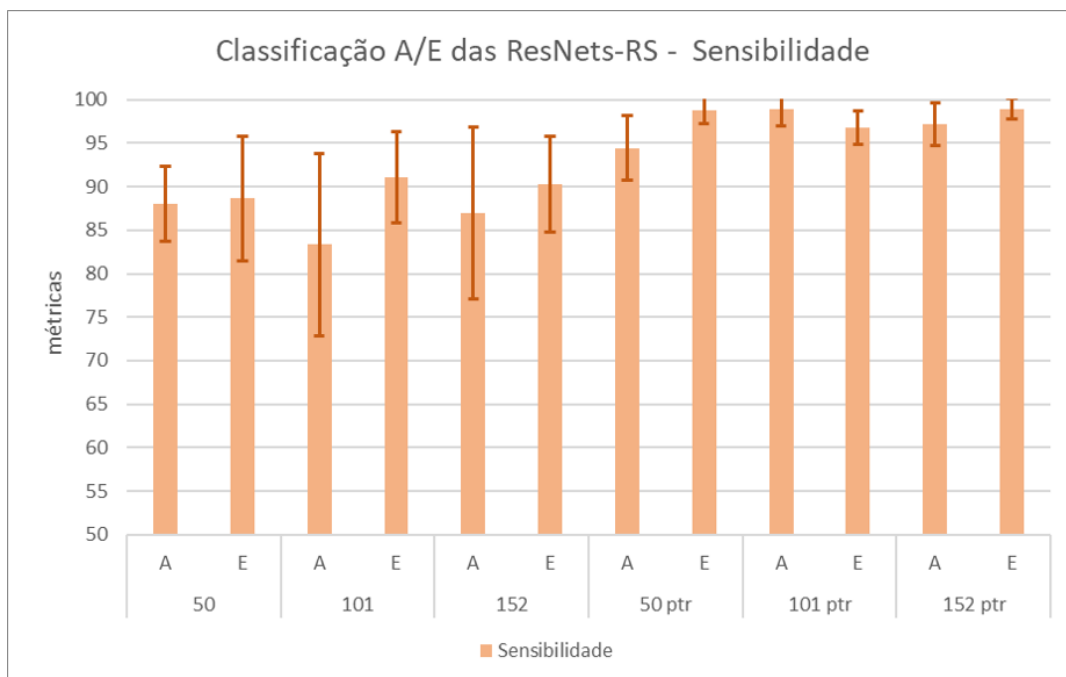


Figura 56 – Comparação dos valores de sensibilidade, em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32.

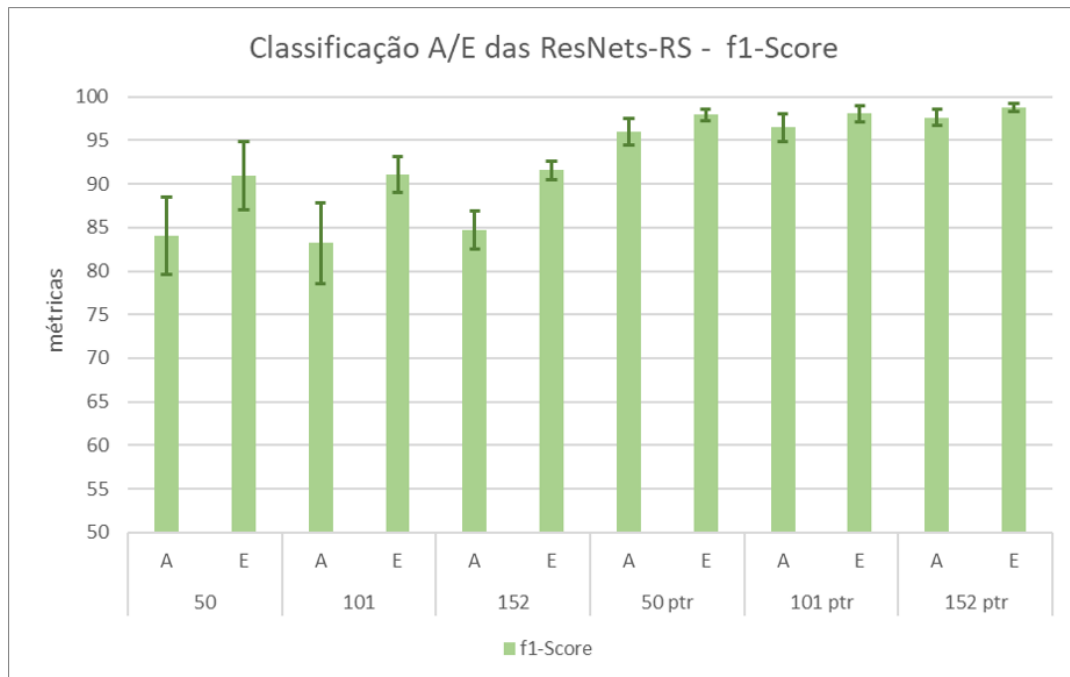


Figura 57 – Comparação dos valores de  $f1$ -score, em tons mais claros, para cada classe, com seus respectivos desvios padrão (em tons mais escuros) para todos os modelos utilizados na classificação das imagens em acumuladoras ou excludentes. Dados obtidos da Tabela 32.

### 4.3.2 Classificação em famílias de plantas

Para esse trabalho, o objetivo da classificação das imagens em famílias de plantas era verificar a facilidade com que os modelos classificavam cada família e comparar com os resultados obtidos das classificações em acumuladoras e excludentes, considerando as características de resistência ao alumínio dessas famílias. Na Tabela 33 são apresentados os dados gerais de acurácias obtidos pelos modelos ResNet-RS utilizados para a classificação das imagens em famílias de plantas. Na Figura 58 é feita a tradução dos dados dessa tabela, com o tempo total convertido em horas para uma melhor visualização dos dados. Vale lembrar que foram utilizadas 20 épocas para os modelos pré-treinados e 200 para os não pré-treinados, o que acaba influenciando nos valores de tempo total dos modelos. Outro fator que influencia na duração do tempo total é o tamanho das arquiteturas, sendo os valores de 50, 101 e 152 das ResNets-RS referentes à quantidade de camadas utilizadas.

Ao se analisar a Tabela 33 e a Figura 58 percebe-se que, ao contrário da classificação binária mostrada anteriormente, não foram as arquiteturas mais complexas que obtiveram maior acurácia. Isso ocorre porque na separação em famílias, espera-se que as plantas categorizadas em uma mesma família específica apresentem características em comum. Sendo assim, as classes de famílias são mais facilmente identificáveis pelas máquinas do que as classes de acumuladoras e excludentes, que agrupam conjuntos de famílias e consequentemente necessitam de uma análise mais profunda da rede neural. Ao se utilizar

Classificação em Famílias - Acurácias Gerais e Desvios Padrão			
ResNet-RS	Acurácia	Desvio Padrão	Tempo Total (min)
50	88,08 %	$\pm 1,23\%$	221 min
101	<b>88,65 %</b>	<b><math>\pm 1,04\%</math></b>	335 min
152	88,56 %	$\pm 2,25\%$	457 min
50 ptr	<b>98,17 %</b>	<b><math>\pm 0,52\%</math></b>	23 min
101 ptr	98,17 %	$\pm 0,67\%$	35 min
152 ptr	97,4 %	$\pm 0,97\%$	50 min

Tabela 33 – Dados resultantes de acurácias gerais, desvios padrão e tempo total para o treino, validação e testes para os 10 *folds* de todos os modelos ResNet-RS utilizados na classificação das imagens de acordo com as famílias das plantas.

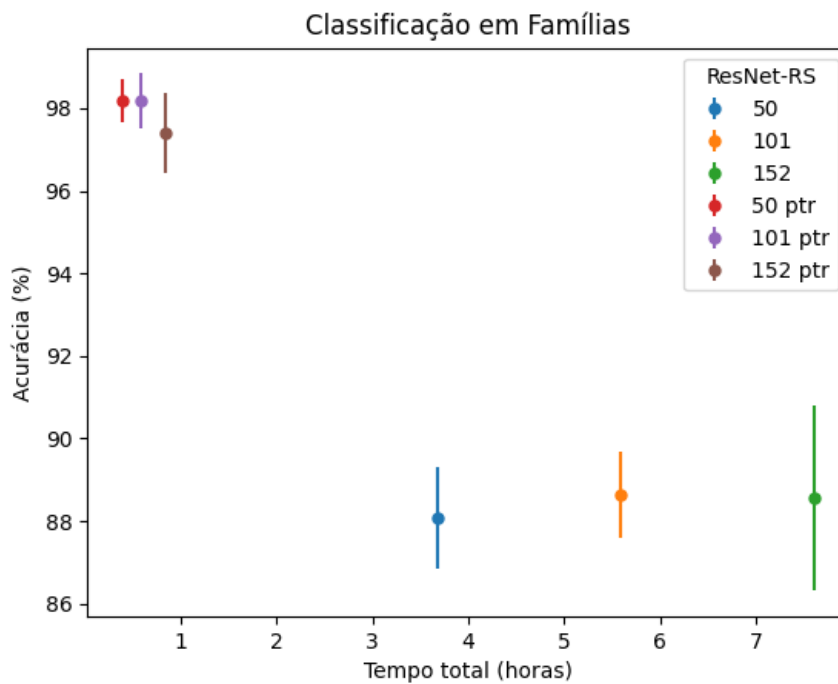


Figura 58 – Tradução dos dados da Tabela 33 comparando as acurácias obtidas com tempo que cada modelo demorou para terminar de rodar. As barras coloridas representam os desvios padrão de cada modelo ResNet-RS.

redes mais profundas para classificações mais "fáceis", os parâmetros em excesso do modelo podem aprender características inerentes ao banco de dados que não são relevantes para a classificação. Isso acaba causando o sobreajuste observado nas curvas de perda das Figuras 42, 44 e 46 dos modelos sem pré-treino, o que pode explicar a grande variação das métricas de desempenho e altos desvios padrão observados nas Tabelas 20, 22 e 24. Em contrapartida, os modelos pré-treinados apresentaram curvas de perda muito boas e altas métricas de desempenho, com desvios padrão relativamente mais baixos, como pode ser observado respectivamente nas Figuras 48, 50 e 52 e nas Tabelas 26, 28 e 30.

Pela quantidade de dados de resultados e pelo objetivo de se analisar a qualidade de classificação para cada classe, optou-se por separar os dados das métricas de desempenho pelas classes de famílias de plantas, tendo sido gerada uma tabela para cada uma das doze famílias apresentadas. As tabelas das métricas de desempenho por modelo, que se encontram no Anexo A, são referentes às famílias: Apocynaceae (Tabela 34), Caryocaraceae (Tabela 35), Celastraceae (Tabela 36), Clusiaceae (Tabela 37), Fabaceae (Tabela 38), Lythraceae (Tabela 39), Malvaceae (Tabela 40), Melastomataceae (Tabela 41), Myrtaceae (Tabela 42), Ochnaceae (Tabela 43), Styracaceae (Tabela 44), Vochysiaceae (Tabela 45). Nesse mesmo anexo também são apresentados os gráficos com os valores das métricas de desempenho de cada classe para cada um dos modelos utilizados, com: a Figura 59 mostrando os valores de precisão, a Figura 60 os valores de sensibilidade e a Figura 61 os valores de *f1-score*.

Verificando a Figura 59 no Anexo A, é possível perceber que os modelos pré-treinados apresentaram, em geral, resultados de precisão mais altos e desvios padrão menores do que os modelos sem pré-treino, tendo o modelo ResNet-RS-50 se destacado nesses quesitos. Na Figura 60, onde são apresentados os valores de sensibilidade, é possível notar essa diferença especialmente para a classe Apocynaceae, que obteve sensibilidade de aproximadamente 60% nos modelos sem pré-treino e 100% nos modelos pré-treinados, como pode também ser verificado na Tabela 34. Como consequência dos resultados de precisão e sensibilidade, os resultados de *f1-score* apresentados na Figura 61, apresentaram comportamento similar: melhores valores gerais obtidos para os modelos pré-treinados do que para os sem pré-treino.

Analisando as tabelas e os gráficos das figuras no Anexo A, percebe-se uma diferença entre as métricas obtidas para os tipos de treino de modelo, muitas vezes obtendo resultados de 100%, com desvios padrão 0%, ou próximos disso nos modelos pré-treinados, como é observado nas famílias Apocynaceae (Tabela 34), Celastraceae (Tabela 36) Fabaceae (Tabela 38), Styracaceae (Tabela 44) e Vochysiaceae (Tabela 45). Foi notado que a classe Clusiaceae (Tabela 37) obteve resultados de 100% de métricas de desempenho para todos os modelos, levando a concluir que essa classe possui características visuais mais facilmente reconhecíveis por redes neurais quando comparada com as demais. Para os resultados observados no modelo ResNet-RS-50 pré treinado, percebe-se que enquanto a maioria das classes obteve resultados mais próximos ou iguais a 100%, a classe da família Caryocaraceae (Tabela 35), obteve uma sensibilidade relativamente mais baixa, com um desvio padrão maior do que os demais. Isso pode ser verificado na matriz de confusão da Figura 49, onde 8 imagens de plantas pertencentes à família Caryocaraceae foram erroneamente classificadas como Vochysiaceae. Os resultados muito altos para a classificação em famílias podem ter ocorrido devido a uma diversidade relativamente baixa de características de imagens para determinadas classes na classificação em famílias. Como um dos principais objetivos da implementação dessa classificação era testar se havia alguma

dificuldade da máquina em se classificar as imagens do banco em famílias para comparar com a classificação binária, os resultados de valores 100% obtidos não atrapalham o objetivo principal do trabalho, mas mostram que a classificação em famílias, para o banco de dados utilizado, pode ser feita em alguns casos com grande facilidade.



## 5 Conclusão

Este trabalho propõe uma análise comparativa do desempenho dos modelos ResNet-Rs-50, 101 e 152 para a classificação das imagens do banco, desenvolvido pelo Laboratório de Nanobiotecnologia da Embrapa Recursos Genéticos e Biotecnologia, entre acumuladoras ou excludentes de alumínio e dentre classes de famílias das plantas. Também é realizada uma análise da qualidade da classificação com o foco em cada classe para os modelos treinados e pré-treinados. Quanto a esse ponto, apesar dos pré-treinos dos parâmetros não terem sido feitos com foco no banco de imagens em questão, pode-se concluir que os modelos pré-treinados no geral, além de necessitarem de menos épocas de treinamento, eles também apresentaram uma média de resultados superior à dos modelos sem pré treino.

Quanto à classificação binária das imagens das plantas em acumuladoras ou excludentes, não se sabe se as redes neurais foram capazes de classificá-las identificando as características inerentes à estratégia de resistência ao alumínio ou se foram encontradas outras características que as duas famílias acumuladoras (*Melastomataceae* e *Vochysiaceae*) possuíam em comum, sendo possível assim agrupá-las em uma mesma classe. Para esclarecimento esse ponto específico, seria preciso uma futura expansão do banco de dados com a adição de mais famílias de plantas acumuladoras e mais espécies em cada família. Nos resultados da classificação em acumuladoras ou excludentes, de modo geral, mesmo os modelos sem pré-treino foram capazes de obter resultados de desempenho gerais (precisão, sensibilidade e *f1-score*) acima de 80% para ambas as classes, o que abre espaço para novas possibilidades em classificação de plantas por imagem microscópicas quanto às suas características fisiológicas gerais ou relacionada ao acúmulo de metais Isso pode auxiliar no desenvolvimento de pesquisas que vão desde a avaliação de plantas comestíveis que, por acumularem determinados metais podem causar problemas de saúde ao ser humano (GUERRA et al., 2012; BOJÓRQUEZ-QUINTAL et al., 2017) até pesquisas relacionadas à fitominação, extração de metais pesados do solo de determinados grupos de plantas acumuladoras, favorecendo também a remediação desses solos (fitorremediação) (AKINBILE; MAKHUBELA; AMBUSHE, 2021; SHEORAN; SHEORAN; POONIA, 2009; ASHRAF et al., 2019).

Como incrementos futuros nesse trabalho, pode-se sugerir:

- Expansão do banco de dados, aumentando a quantidade e variedade de imagens por classe, principalmente para as acumuladoras;
- Possibilidade de adaptação do programa em outros bancos de dados para detecção de outros metais presentes no tecido vegetal, como chumbo, níquel, ouro, mercúrio, etc;

- Implementação do programa em ambiente de fácil utilização por um usuário, como por exemplo: programa com interface, aplicativo móvel, etc.;
- Adição de novas classes para novas possibilidades de classificação, como por exemplo: "acumuladoras", "excludentes" e "não resistentes" à presença do metal;
- Implementação da classificação levando em consideração a quantidade de alumínio acumulado pela planta;
- Implementação da visualização das características específicas identificadas para cada classe (Podendo ter grande possibilidade da imagem resultante não ser humanamente interpretável).

## Referências

- AKINBILE, B. J.; MAKHUBELA, B. C. E.; AMBUSHE, A. A. Phytomining of valuable metals: status and prospective-a review. *International Journal of Environmental Analytical Chemistry*, Taylor Francis, v. 0, n. 0, p. 1–21, 2021. Disponível em: <<https://doi.org/10.1080/03067319.2021.1917557>>. Citado 3 vezes nas páginas 10, 12 e 95.
- ANZANELLO, M. J.; FOGLIATTO, F. S. Learning curve models and applications: Literature review and research directions. *International Journal of Industrial Ergonomics*, v. 41, n. 5, p. 573–583, 2011. ISSN 0169-8141. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S016981411100062X>>. Citado na página 52.
- ASHRAF, S. et al. Phytoremediation: Environmentally sustainable way for reclamation of heavy metal polluted soils. *Ecotoxicology and Environmental Safety*, v. 174, p. 714–727, 2019. Citado 3 vezes nas páginas 10, 12 e 95.
- BELLO, I. et al. Revisiting resnets: Improved training and scaling strategies. *arXiv preprint arXiv:2103.07579*, 2021. Citado 5 vezes nas páginas 17, 49, 55, 56 e 60.
- BERRAR, D. Cross-validation. In: \_\_\_\_\_. [S.l.: s.n.], 2018. ISBN 9780128096338. Citado na página 43.
- BISHOP, C. M. Pattern recognition and machine learning. In: . [S.l.: s.n.], 2006. ISBN 0-387-31073-8. Citado na página 35.
- BOJÓRQUEZ-QUINTAL, E. et al. Aluminum, a Friend or Foe of Higher Plants in Acid Soils. *Frontiers in Plant Science*, v. 8, 2017. Citado 3 vezes nas páginas 10, 12 e 95.
- CONCEIÇÃO, L. D. H. C. S. D.; SERENO, M. J. C. D. M.; NETO, J. F. B. Tolerância ao alumínio em plantas: toxicidade, mecanismos e genes em espécies cultivadas. *Revista Brasileira de Agrociência*, Pelotas, v. 14, n. 3-4, p. 1–10, 2008. Citado 2 vezes nas páginas 29 e 33.
- COTTA, M. et al. Diferenças entre hematoxilina e aluminon na detecção de alumínio em tecidos foliares de plantas nativas do cerrado. In: FALEIRO, F.; NETO, A. F. (Ed.). Brasília, Brasil: [s.n.], 2008. Citado na página 30.
- CUBUK, E. D. et al. *RandAugment: Practical automated data augmentation with a reduced search space*. 2019. Citado na página 45.
- DERAKHSHAN, B. et al. Optimizing machine learning workloads in collaborative environments. In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: Association for Computing Machinery, 2020. (SIGMOD '20), p. 1701–1716. ISBN 9781450367356. Disponível em: <<https://doi.org/10.1145/3318464.3389715>>. Citado na página 58.
- ECHART, C. L.; CAVALLI-MOLINA, S. Fitotoxicidade do alumínio: efeitos, mecanismo de tolerância e seu controle genético. *Ciência Rural*, v. 31, n. 3, Jun 2001. Disponível em: <<https://doi.org/10.1590/S0103-84782001000300030>>. Citado na página 33.

- FURTADO, M. I. V. Redes neurais artificiais: Uma abordagem para sala de aula. In: . [S.l.]: Atena Editora, 2019. Citado 2 vezes nas páginas 35 e 36.
- GALLO, C. Artificial neural networks tutorial. *Encyclopedia of Information Science and Technology*, Information science Reference, Janeiro 2015. Citado na página 36.
- GIANEY, H. K.; CHOUDHARY, R. Comprehensive review on supervised machine learning algorithms. In: IEEE. *2017 International Conference on Machine Learning and Data Science (MLDS)*. [S.l.], 2017. p. 37–43. Citado na página 49.
- GONZALEZ, R. C.; WOODS, R. E. In: *Processamento Digital de Imagens*. [S.l.]: Pearson, 2009. v. 3. Citado na página 33.
- GUERRA, F. et al. Heavy metals in vegetables and potential risk for human health. *Scientia Agricola*, v. 69, n. 1, p. 54–60, 2012. Citado 3 vezes nas páginas 10, 12 e 95.
- HAN, C. et al. Learning smooth representations with generalized softmax for unsupervised domain adaptation. *Information Sciences*, v. 544, p. 415–426, 2021. ISSN 0020-0255. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0020025520308458>>. Citado na página 39.
- HARIDASAN, M. Nutritional adaptations of native plants of the cerrado biome in acid soils. *Brazilian Journal of Plant Physiology*, Universidade de Brasília, Instituto de Ciências Biológicas, Departamento de Ecologia, v. 20, n. 3, p. 183–195, Outubro 2008. Citado 3 vezes nas páginas 29, 30 e 33.
- HE, K. et al. *Deep Residual Learning for Image Recognition*. 2015. Citado na página 46.
- HINTON, G. E. et al. *Improving neural networks by preventing co-adaptation of feature detectors*. 2012. Citado na página 42.
- HU, J. et al. *Squeeze-and-Excitation Networks*. 2019. Citado na página 48.
- HUANG, G. et al. *Deep Networks with Stochastic Depth*. 2016. Citado na página 49.
- INDOLIA, S. et al. Conceptual understanding of convolutional neural network- a deep learning approach. *Procedia Computer Science*, Elsevier Ltd., p. 680–688, 2018. Citado 3 vezes nas páginas 37, 38 e 42.
- IOFFE, S.; SZEGEDY, C. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. Citado na página 41.
- JÄHNE, B. In: *Digital Image Processing*. [S.l.]: Springer-Verlag, 1995. v. 3. Citado na página 33.
- KHAN, S. et al. In: MEDIONI, G.; DICKINSON, S. (Ed.). *A Guide to Convolutional Neural Networks for Computer Vision*. [S.l.]: Morgan and Claypool Publishers, 2018. ISBN 9781681730226. ISSN c 2153-1064. Citado 9 vezes nas páginas 13, 37, 38, 39, 40, 41, 44, 45 e 46.
- KOHAVI, R.; PROVOST, F. Glossary of terms. *Machine Learning*, v. 30, p. 271–274, 1998. Citado na página 35.
- LECUN, Y. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, MIT Press, v. 1, n. 4, p. 541–551, 1989. Citado na página 37.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998. Citado na página 39.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *Journal of Materials Research and Technology*, v. 5, n. 4, p. 115–133, 1943. ISSN 1522-9602. Citado na página 36.

MOHR, F.; RIJN, J. N. van. *Learning Curves for Decision Making in Supervised Machine Learning – A Survey*. arXiv, 2022. Disponível em: <<https://arxiv.org/abs/2201.12150>>. Citado na página 52.

OEI, R. W. et al. Convolutional neural network for cell classification using microscope images of intracellular actin networks. *PLOS ONE*, Public Library of Science, v. 14, n. 3, p. 1–13, 03 2019. Disponível em: <<https://doi.org/10.1371/journal.pone.0213626>>. Citado na página 29.

OH, K.-S.; JUNG, K. Gpu implementation of neural networks. *Pattern Recognition*, v. 37, n. 6, p. 1311–1314, 2004. ISSN 0031-3203. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0031320304000524>>. Citado na página 37.

OLIVEIRA, A. P. de et al. Avaliação de espécies do cerrado quanto à tolerância ao alumínio. *Boletim de Pesquisa e Desenvolvimento*, Embrapa, n. 355, Dezembro 2019. ISSN 0102-0110. Citado 5 vezes nas páginas 17, 29, 30, 33 e 54.

PANDA, S. K.; MATSUMOTO, H. Molecular physiology of aluminum toxicity and tolerance in plants. *Botanical Review*, Springer on behalf of New York Botanical Garden Press, v. 73, n. 4, p. 326–347, Oct. - Dec. 2007. Citado 2 vezes nas páginas 29 e 30.

PERLICH, C. Learning curves in machine learning. p. 1–5, 01 2011. Citado na página 52.

QUEIROZ, J. E. R. de; GOMES, H. M. Introdução ao processamento digital de imagens. *Revista RITA*, v. 13, n. 1, 2001. Citado na página 33.

REVERDY, P.; LEONARD, N. E. *Parameter estimation in softmax decision-making models with linear objective functions*. 2015. Citado na página 39.

ROUT, G.; SAMANTARAY, S.; DAS, P. Aluminium toxicity in plants: a review. *Agronomie*, v. 21, n. 1, p. 3–21, 2001. 10.1051/agro:2001105. Id: hal-00886101. Citado na página 33.

RUSSAKOVSKY, O. et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, v. 115, n. 3, p. 211–252, 2015. Citado na página 30.

RUSSEL, S.; NORVIG, P. In: *Artificial Intelligence: A Modern Approach*. [S.l.: s.n.], 2003. ISBN 978-0137903955. Citado na página 35.

SANDLER, M. et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. Citado na página 48.

SHEORAN, V.; SHEORAN, A.; POONIA, P. Phytomining: A review. *Minerals Engineering*, v. 22, n. 12, p. 1007–1019, 2009. ISSN 0892-6875. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S089268750900106X>>. Citado 3 vezes nas páginas 10, 12 e 95.

- SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. Citado 2 vezes nas páginas 45 e 46.
- SRIVASTAVA, N. et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, v. 15, n. 56, p. 1929–1958, 2014. Disponível em: <<http://jmlr.org/papers/v15/srivastava14a.html>>. Citado 3 vezes nas páginas 13, 40 e 41.
- SUN, Y. et al. Deep Learning for Plant Identification in Natural Environment. *Computational Intelligence and Neuroscience*, v. 2017, p. 1–6, 2017. Citado na página 31.
- SUNNY, N. M. A Review on Deep Learning for Plant Species Classification using Leaf Vein. *International Journal of Engineering Research and*, V9, n. 06, 2020. Citado na página 31.
- TAN, M.; LE, Q. V. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. 2020. Citado 2 vezes nas páginas 13 e 47.
- TAN, M.; LE, Q. V. *EfficientNetV2: Smaller Models and Faster Training*. 2021. Citado na página 48.
- VITORELLO, V. A.; CAPALDI, F. R.; STEFANUTO, V. A. Recent advances in aluminum toxicity and resistance in higher plants. *Brazilian Journal of Plant Physiology*, v. 17, p. 43–129, Mar 2005. Disponível em: <<https://doi.org/10.1590/S1677-04202005000100011>>. Citado na página 29.
- WANG, A. Y. et al. What makes a well-documented notebook? a case study of data scientists' documentation practices in kaggle. In: \_\_\_\_\_. New York, NY, USA: Association for Computing Machinery, 2021. ISBN 9781450380959. Disponível em: <<https://doi.org/10.1145/3411763.3451617>>. Citado na página 58.
- WISTUBA, M.; PEDAPATI, T. Learning to rank learning curves. In: III, H. D.; SINGH, A. (Ed.). *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020. (Proceedings of Machine Learning Research, v. 119), p. 10303–10312. Disponível em: <<https://proceedings.mlr.press/v119/wistuba20a.html>>. Citado na página 52.
- YAMASHITA, R.; NISHIO, M.; DO, R. Convolutional neural networks: an overview and application in radiology. v. 9, p. 611–629, 2018. Citado 3 vezes nas páginas 13, 41 e 42.
- YUAN, L. et al. *VOLO: Vision Outlooker for Visual Recognition*. 2021. Citado na página 57.

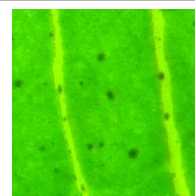
# Anexos





## ANEXO A – Resultados Gerais: Métricas de Desempenho das Classificações em Famílias

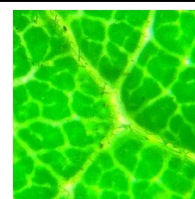
Métricas de Desempenho e Desvios Padrão  
Classe 00: Apocynaceae - Excludente (52 imagens)



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	88,33 %	±14,63%	60%	±0%	69%	±5,19%
<b>101</b>	87,88 %	±14,28%	58%	±6%	69,88%	±6,61%
<b>152</b>	82,35 %	±13,15%	56%	±8%	66,67 %	±7,59%
<b>50 ptr</b>	100 %	±0%	100 %	±0%	100 %	±0%
<b>101 ptr</b>	100 %	±0%	100 %	±0%	100 %	±0%
<b>152 ptr</b>	100 %	±0%	100 %	±0%	100 %	±0%

Tabela 34 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Apocynaceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Valores obtidos como médias de todos os  *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Hancornia speciosa*, da família Apocynaceae.

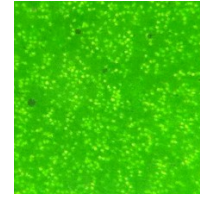
Métricas de Desempenho e Desvios Padrão  
Classe 01: Caryocaraceae - Excludente (52 imagens)



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	89,36 %	±9,4%	84%	±8%	86,6 %	±4,4%
<b>101</b>	93,48 %	±8,7%	86 %	±9,17%	89,58 %	±6,36%
<b>152</b>	95,56 %	±6,67%	86 %	±9,17%	90,53 %	±3,3%
<b>50 ptr</b>	100 %	±0%	84 %	±8%	91,3 %	±4,44%
<b>101 ptr</b>	98,04 %	±5%	100 %	±0%	99,01 %	±2,73%
<b>152 ptr</b>	100 %	±0%	100 %	±0%	100 %	±0%

Tabela 35 – Dados resultantes das métricas de desempenho e desvios padrão para a classe Caryocaraceae com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os  *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Caryocar brasiliense*, da família Caryocaraceae.

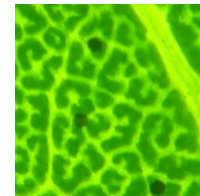
**Métricas de Desempenho e Desvios Padrão**  
**Classe 02: Celastraceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	100 %	±0%	74 %	±18%	85 %	±11,3%
<b>101</b>	100 %	±0%	86 %	±18%	92,47 %	±11,21%
<b>152</b>	100 %	±0%	76 %	±17,44%	86,36 %	±10,97%
<b>50 ptr</b>	100 %	±0%	100%	±0%	100 %	±0%
<b>101 ptr</b>	100 %	±0%	100 %	±0%	100 %	±0%
<b>152 ptr</b>	100 %	±0%	98 %	±6%	98,99 %	±3,33%

Tabela 36 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Celastraceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Salacia crassifolia*, da família *Celastraceae*.

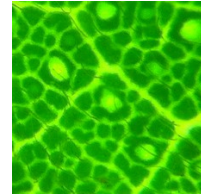
**Métricas de Desempenho e Desvios Padrão**  
**Classe 03: Clusiaceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	100%	±0%	100 %	±0%	100 %	±0%
<b>101</b>	100%	±0%	100 %	±0%	100 %	±0%
<b>152</b>	100%	±0%	100 %	±0%	100 %	±0%
<b>50 ptr</b>	100%	±0%	100 %	±0%	100 %	±0%
<b>101 ptr</b>	100%	±0%	100 %	±0%	100 %	±0%
<b>152 ptr</b>	100%	±0%	100 %	±0%	100 %	±0%

Tabela 37 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Clusiaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Kielmeyera coriacea*, da família *Clusiaceae*.

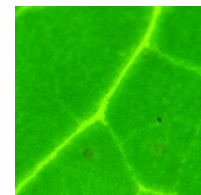
**Métricas de Desempenho e Desvios Padrão**  
**Classe 04: Fabaceae - Excludente (208 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	85,1 %	±5, 85%	84,29 %	±4, 29%	84,69 %	±3, 48%
<b>101</b>	82,35%	±4, 55%	86,67%	±4, 15%	84,45%	±3, 03%
<b>152</b>	81,13%	±4, 92%	89,52%	±6, 67%	85,07%	±4, 16%
<b>50 ptr</b>	95,02 %	±2, 32%	100%	±0%	97,45%	±1, 22%
<b>101 ptr</b>	94,59%	±3, 66%	100%	±0%	97,22%	±1, 94%
<b>152 ptr</b>	97,14%	±3, 03%	97,14%	±3, 16%	97,14%	±2, 07%

Tabela 38 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Fabaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Hymenaea stigonocarpa*, da família *Fabaceae*.

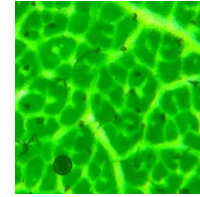
**Métricas de Desempenho e Desvios Padrão**  
**Classe 05: Lythraceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	90,7%	±9, 43%	78%	±10, 77%	83,87%	±5, 93%
<b>101</b>	92,86%	±8, 7%	78%	±10, 77%	84,78%	±5, 99%
<b>152</b>	93,02%	±11, 08%	80%	±0%	86,02%	±5, 27%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	100%	±0%	92%	±9, 8%	95,83%	±5, 44%
<b>152 ptr</b>	98,04%	±5%	100%	±0%	99,01%	±2, 73%

Tabela 39 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Lythraceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folds*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Lafoensia pacari*, da família *Lythraceae*.

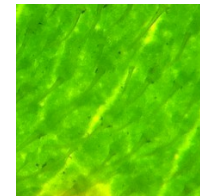
Métricas de Desempenho e Desvios Padrão  
Classe 06: Malvaceae - Excludente (52 imagens)



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	95,35 %	±8,57%	82 %	±6%	88,17 %	±1,67%
<b>101</b>	97,62 %	±5%	82 %	±6%	89,13 %	±0,61%
<b>152</b>	100 %	±0%	82 %	±6%	90,11 %	±3,33%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>152 ptr</b>	100 %	±0%	92 %	±9,8%	95,83 %	±5,44%

Tabela 40 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Malvaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Eriotheca gracilipes*, da família *Malvaceae*.

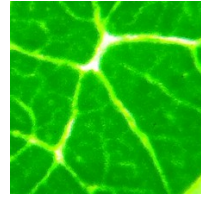
Métricas de Desempenho e Desvios Padrão  
Classe 07: Melastomataceae - Acumuladora (104 imagens)



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	82,71 %	±6,36%	100 %	±0%	90,53%	±3,92%
<b>101</b>	82,58 %	±6,9%	99,09 %	±2,73%	90,08%	±4,58%
<b>152</b>	83,08 %	±5,03%	98,18 %	±3,64%	90%	±3,71%
<b>50 ptr</b>	100 %	±0%	91,82 %	±2,73%	95,73%	±1,43%
<b>101 ptr</b>	99,06 %	±2,73%	95,45 %	±4,55%	97,22%	±3,07%
<b>152 ptr</b>	92,24 %	±2,6%	97,27 %	±4,17%	94,69%	±2,77%

Tabela 41 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Melastomataceae* com imagens de plantas acumuladoras para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Pleroma stenocarpum*, da família *Melastomataceae*.

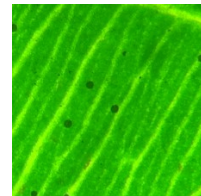
**Métricas de Desempenho e Desvios Padrão**  
**Classe 08: Myrtaceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	84,75 %	±10,24%	100 %	±0%	91,74%	±5,9%
<b>101</b>	94,34 %	±7,64%	100 %	±0%	97,09%	±4,17%
<b>152</b>	83,33 %	±12,81%	100 %	±0%	90,91%	±7,46%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	90,91 %	±8,33%	100 %	±0%	95,24%	±4,55%
<b>152 ptr</b>	87,72 %	±11,48%	100 %	±0%	93,46%	±6,6%

Tabela 42 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Myrtaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Psidium myrsinoides*, da família *Myrtaceae*.

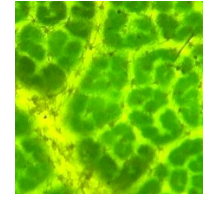
**Métricas de Desempenho e Desvios Padrão**  
**Classe 09: Ochnaceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	89,29 %	±8,16%	83,33 %	±0%	86,21 %	±3,71%
<b>101</b>	89,9 %	±11,7%	81,67 %	±5%	85,22 %	±5,77%
<b>152</b>	91,23 %	±9,98%	86,67 %	±6,67%	88,89 %	±6%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	100 %	±0%	83,33 %	±0%	90,91 %	±0%
<b>152 ptr</b>	100 %	±0%	83,33 %	±0%	90,91 %	±0%

Tabela 43 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Ochnaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Ouratea hexasperma*, da família *Ochnaceae*.

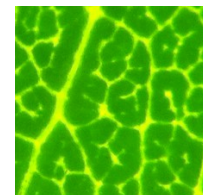
**Métricas de Desempenho e Desvios Padrão**  
**Classe 10: Styracaceae - Excludente (52 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	100 %	±0%	98 %	±6%	98,99 %	±3,33%
<b>101</b>	98 %	±5%	98 %	±6%	98 %	±4,07%
<b>152</b>	100 %	±0%	98 %	±6%	98,99 %	±3,33%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>152 ptr</b>	100%	±0%	100%	±0%	100%	±0%

Tabela 44 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Styracaceae* com imagens de plantas excludentes para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Styrax ferrugineus*, da família *Styracaceae*.

**Métricas de Desempenho e Desvios Padrão**  
**Classe 11: Vochysiaceae - Acumuladora (260 imagens)**



ResNet-RS	Precisão e DP		Sensibilidade e DP		f1-Score e DP	
<b>50</b>	100 %	±0%	98 %	±6%	98,99 %	±3,33%
<b>101</b>	98 %	±5%	98 %	±6%	98 %	±4,07%
<b>152</b>	100 %	±0%	98 %	±6%	98,99 %	±3,33%
<b>50 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>101 ptr</b>	100%	±0%	100%	±0%	100%	±0%
<b>152 ptr</b>	100%	±0%	100%	±0%	100%	±0%

Tabela 45 – Dados resultantes das métricas de desempenho e desvios padrão para a classe *Vochysiaceae* com imagens de plantas acumuladoras para todos os modelos ResNet-RS utilizados. Dados obtidos dos valores de todos os *folders*. A folha da imagem na parte direita superior foi retirada do banco de dados e pertence à espécie *Qualea parviflora*, da família *Vochysiaceae*.

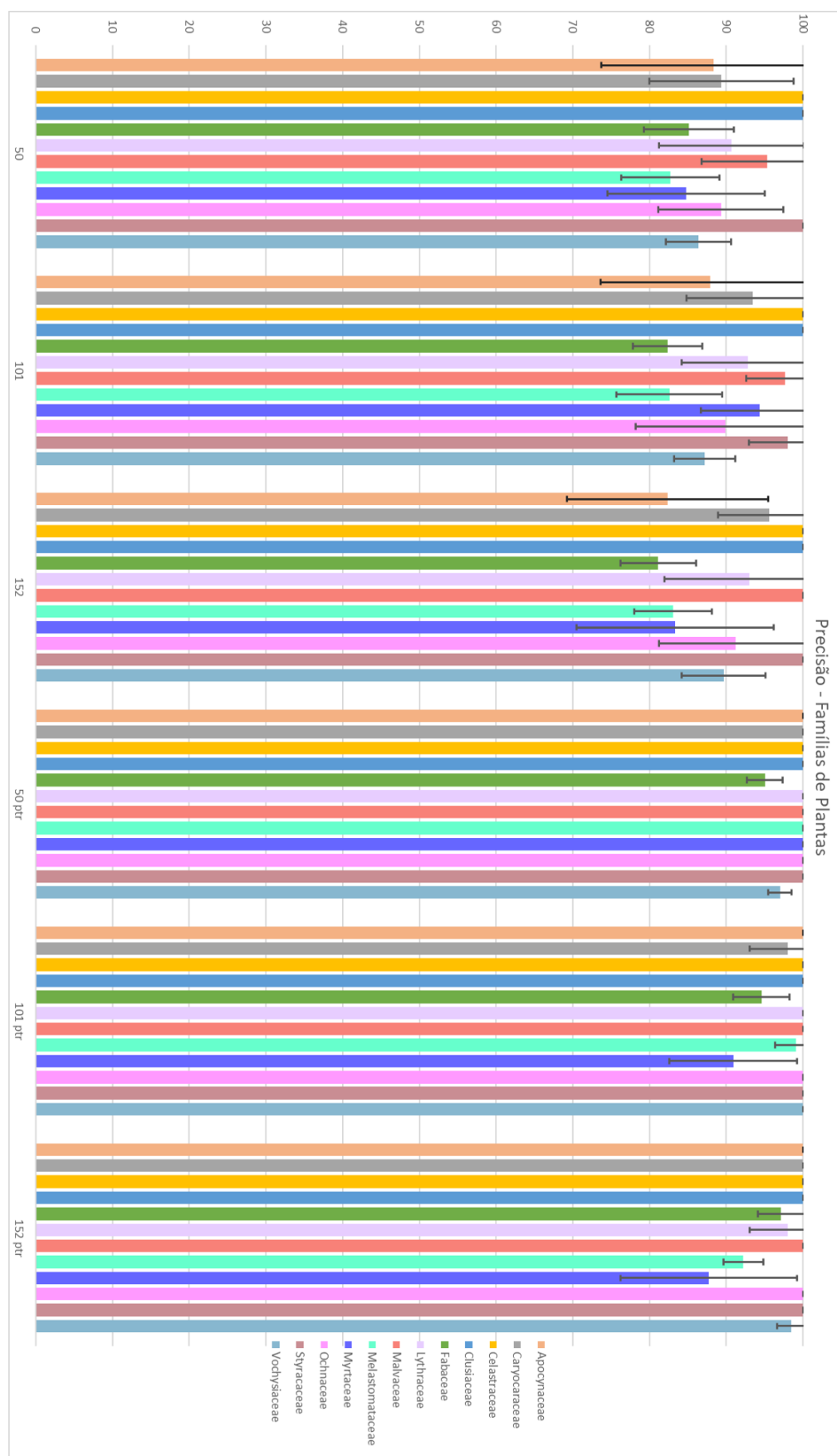


Figura 59 – Valores de precisão obtidos para em cada modelo ResNet-RS para cada família de planta.

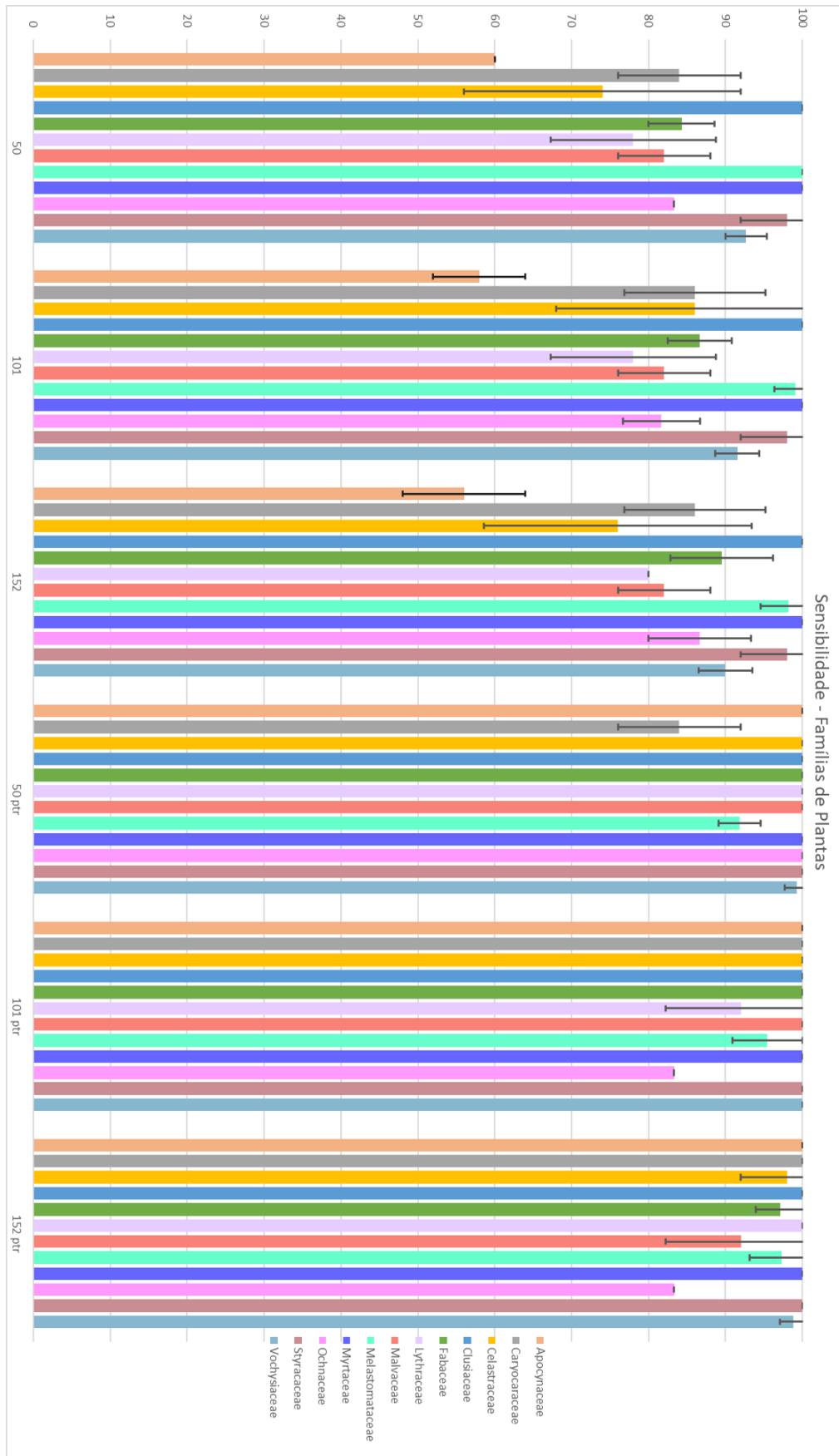


Figura 60 – Valores de sensibilidade obtidos para em cada modelo ResNet-RS para cada família de planta.



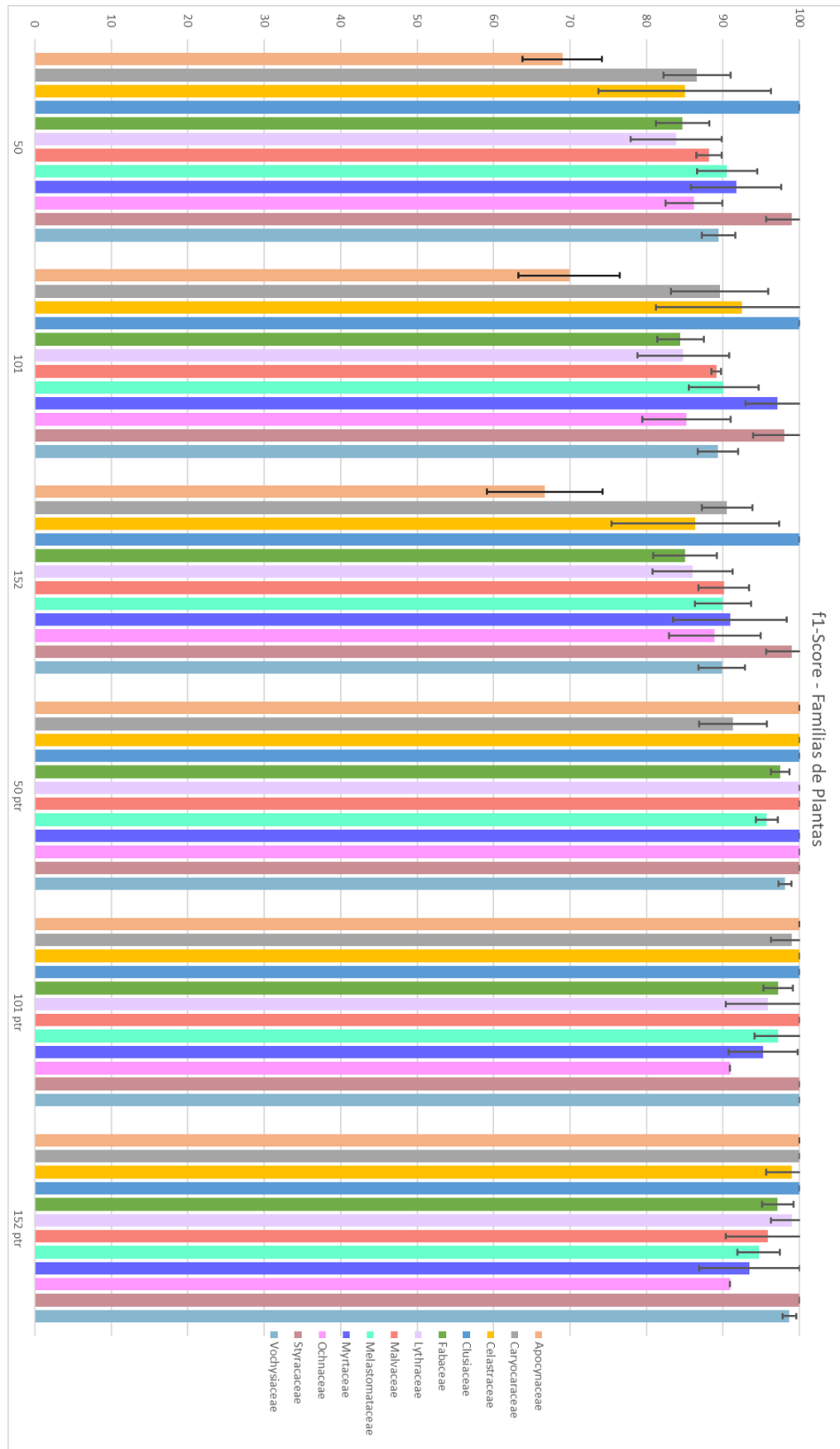


Figura 61 – Valores de  $f1$ -score obtidos para em cada modelo ResNet-RS para cada família de planta.