

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROJETO DE GRADUAÇÃO**



ANTONIO DRAGO CAETANO

**DETECÇÃO DE CÂNCER DE MAMA PELA APLICAÇÃO
DE REDES NEURAIIS PROFUNDAS EM IMAGENS**

VITÓRIA
2022

ANTONIO DRAGO CAETANO

**DETECÇÃO DE CÂNCER DE MAMA PELA APLICAÇÃO DE REDES
NEURAIS PROFUNDAS EM IMAGENS**

Parte manuscrita do Projeto de Graduação do aluno **Antonio Drago Caetano**, apresentada ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para a obtenção do grau de Engenheiro Eletricista.

Orientador: Prof. Dr. Patrick Marques Ciarelli

VITÓRIA
2022

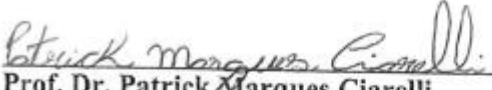
ANTONIO DRAGO CAETANO

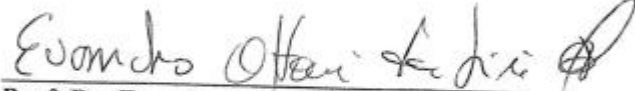
**DETECÇÃO DE CÂNCER DE MAMA PELA APLICAÇÃO DE REDES
NEURAIIS PROFUNDAS EM IMAGENS**

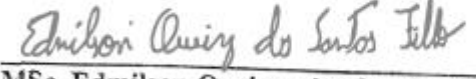
Parte manuscrita do Projeto de Graduação do
aluno **Antonio Drago Caetano**, apresentado
ao Departamento de Engenharia Elétrica do
Centro Tecnológico da Universidade Federal
do Espírito Santo, como requisito parcial para
obtenção do grau de Engenheiro Eletricista.

Aprovada em 18 de agosto de 2022.

COMISSÃO EXAMINADORA:


Prof. Dr. Patrick Marques Ciarelli
Universidade Federal do Espírito Santo
Orientador


Prof. Dr. Evandro Ottoni Teatini Salles
Universidade Federal do Espírito Santo
Examinador


MSc. Edmilson Queiroz dos Santos Filho
Examinador

A todos amigos e familiares que estiveram ao meu lado durante a graduação.

AGRADECIMENTOS

Agradeço a minha família por sempre ter colocado minha educação como prioridade, por servirem de exemplo de dedicação e por todo amor demonstrado ao longo desta jornada.

À minha namorada Larissa por todo apoio e sempre me motivar a ser uma pessoa melhor a cada dia.

Agradeço ao meu orientador Prof. Dr. Patrick Marques Ciarelli por toda paciência e ensinamentos sempre se pondo a disposição de tirar dúvidas e me tranquilizar nos momentos de desespero.

A todos meus amigos do curso de Engenharia Elétrica que sempre estiveram ao lado nos momentos mais difíceis da graduação dividindo os medos e comemorando as conquistas.

Aos meus colegas de trabalho e à EDP Espírito Santo por todo incentivo.

À banca examinadora por ter aceitado o convite de avaliar o presente trabalho.

Por fim, à Universidade Federal do Espírito Santo e o Departamento de Engenharia Elétrica pela minha graduação.

RESUMO

O câncer de mama é um dos cânceres mais comum do mundo segundo o instituto americano de pesquisa sobre o câncer (do inglês, *World Cancer Research Fund International* – WCRF). Em 2020, o câncer de mama foi responsável por 12,5% do número total de diagnósticos da doença. A detecção precoce é uma das formas mais eficazes de se combater esta doença responsável pela perda de tantas vidas femininas ao longo dos anos. Com o avanço da tecnologia, a detecção e tratamento de doenças precoces têm se tornado cada vez mais eficientes e precisos, trazendo maior assertividade no diagnóstico deixando o tratamento menos extremo e invasivo. Este trabalho tem como objetivo aplicar técnicas de aprendizado de máquina, mais especificamente, técnicas de redes neurais profundas, para a predição de câncer de mama em um conjunto de imagens. Para isso, foi utilizada uma base de dados pública que possui diversas microscopias de tecido mamário com a presença ou não de tecido cancerígeno. Três técnicas foram avaliadas para a tarefa de detecção de câncer nas imagens de microscopia: redes neurais convolucionais, *Support Vector Machines* (SVM) e uma técnica baseada em árvore de decisão que é otimizada por gradiente descendente, chamada *Extreme Gradient Boosting* (XGBoost). Inicialmente, duas arquiteturas de redes neurais convolucionais, especificamente VGG-19 e ResNet-50, foram avaliadas para extrair características das imagens e classificá-las, usando técnicas de *transfer learning* e aumento de dados. Depois, a rede com melhor desempenho foi escolhida para extrair características das imagens que foram usadas para treinar e avaliar as outras duas técnicas. Como resultado, foram obtidas acurácias de 80% e 85% para as redes VGG-19 e ResNet-50, respectivamente. As técnicas XGBoost e SVM alcançaram acurácias de 72,5% e 87,5%, respectivamente. Os resultados indicam que o uso de redes convolucionais como extratores de características pode ser uma abordagem viável para o treinamento de classificadores clássicos, sendo os resultados alcançados compatíveis com os melhores obtidos para a mesma base de dados.

Palavras-chave: Aprendizado Profundo. Câncer de mama. Redes Neurais Convolucionais. Classificação de imagens.

ABSTRACT

Breast cancer is one of the most common cancers in the world according to the World Cancer Research Fund International (WCRF). In 2020, breast cancer was responsible for 12.5% of the total number of diagnoses of the disease. Early detection is one of the most effective ways to combat this disease responsible for the loss of female lives over the years. With the advancement of technology, the detection and treatment of early diseases have become increasingly efficient and accurate, bringing greater assertiveness in the diagnosis, leaving the treatment less extreme and invasive. This work aims to apply machine learning techniques, more specifically, deep neural network techniques, for the prediction of breast cancer in a set of images. For this, we used a public database that has several microscopies of breast tissue with the presence or not of cancer tissue. Three techniques were evaluated for the task of detecting cancer in microscopy images: Convolutional Neural Networks (CNNs), Support Vector Machines (SVM) and a decision tree-based technique that is optimized by gradient descent, called Extreme Gradient Boosting (XGBoost). Initially, two convolutional neural network architectures, specifically VGG-19 and ResNet-50, were evaluated to extract features from images and classify them, using transfer learning and data augmentation techniques. Then, the network with the best performance was chosen to extract features from the images that were used to train and evaluate the other two techniques. As a result, accuracies of 80% and 85% were obtained for the VGG-19 and ResNet-50 networks, respectively. The XGBoost and SVM techniques reached accuracies of 72.5% and 87.5%, respectively. The results indicate that the use of convolutional networks as feature extractors can be a viable approach for training classical classifiers, although to achieve better results it would be necessary to evaluate other approaches. The results indicate that the use of convolutional networks as feature extractors can be a viable approach for training classical classifiers, being the achieved results compatible with the best results obtained for the same database.

Keywords: Deep Learning. Breast cancer. Convolutional Neural Networks. Image classification.

LISTA DE FIGURAS

Figura 1 – Exemplo de uma reta separando duas classes	17
Figura 2 – Margens de um Hiperplano	18
Figura 3 – Método Todos-Contra-Todos para 4 classes.....	19
Figura 4 – Exemplo de árvore de decisão.....	20
Figura 5 – Exemplo de uma CNN.	21
Figura 6 – Camada de convolução pra obtenção de características.....	22
Figura 7 – Exemplo de <i>pooling</i>	23
Figura 8 – Camada totalmente conectada.....	23
Figura 9 – Atalhos entre camadas.....	27
Figura 10 – Comparação entre VGG-19 e ResNet-34.....	29
Figura 11 – Exemplo de matriz de confusão com 4 classes.	31
Figura 12 – Análise de métricas para a Classe C_2	32
Figura 13 – Exemplo das imagens microscópicas. A – Normal, B – Benigno, C – Câncer não invasivo, D – Câncer invasivo	36
Figura 14 – Exemplo de imagens geradas pelo <i>ImageDataGenerator</i>	38
Figura 15 – Variação da taxa de aprendizado ao longo das épocas da função <i>Cosine Decay Restart</i>	40
Figura 16 – Matriz de confusão VGG-19	45
Figura 17 – Matriz de confusão ResNet-50	47
Figura 18 – Matriz de confusão ResNet-50 + SVM	49
Figura 19 – Matriz de confusão ResNet-50 + XGBoost	51

LISTA DE TABELAS

Tabela 1 – Estruturas da rede VGG-19.....	26
Tabela 2 – Configurações da rede ResNet.....	28
Tabela 3 – Métricas por classe para a rede VGG-19	44
Tabela 4 – Métricas Macro para a rede VGG-19	44
Tabela 5 – Métricas por classe para a rede ResNet-50.....	46
Tabela 6 – Métricas Macro para a rede ResNet-50	46
Tabela 7 – Métricas por classe para o modelo ResNet-50 + SVM.....	48
Tabela 8 – Métricas Macro para a rede ResNet-50 + SVM	48
Tabela 9 – Métricas por classe para o modelo ResNet-50 + XGBoost	50
Tabela 10 – Métricas Macro para a rede ResNet-50 + XGBoost.....	50
Tabela 11 – Comparação dos modelos de classificação.....	52
Tabela 12 – Comparação do melhor modelo obtido com top-5 resultados do desafio.....	53

LISTA DE ABREVIATURAS E SIGLAS

BACH	<i>Grand Challenge on Breast Cancer Histology Images</i>
CNN	<i>Convolutional Neural Network</i>
FN	<i>False Negative</i>
FP	<i>False Positive</i>
GPU	<i>Graphics Processing Unit</i>
GBM	<i>Gradient Boosting Model</i>
ICIAR	<i>International Conference on Image Analysis and Recognition</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
ILSVRC	<i>ImageNet Large Scale Visual Recognition Challenge</i>
INCA	<i>Instituto Nacional de Câncer</i>
PPV	<i>Positive Predictive Value</i>
PCA	<i>Principal Components Analysis</i>
SGD	<i>Stochastic Gradient Descent</i>
SVM	<i>Support Vector Machine</i>
TN	<i>True Negative</i>
TNR	<i>True Negative Rate</i>
TP	<i>True Positive</i>
TPR	<i>True Positive Rate</i>
UFES	<i>Universidade Federal do Espírito Santo</i>
WHO	<i>World Health Organization</i>

SUMÁRIO

1	INTRODUÇÃO.....	13
1.1	Justificativas.....	14
1.2	Objetivos.....	15
1.2.1	Objetivo Geral	15
1.2.2	Objetivos específicos	15
1.3	Organização do Trabalho	16
2	REFERENCIAL TEÓRICO.....	17
2.1	<i>Support Vector Machines</i>	17
2.2	<i>Extreme Gradient Boosting</i>	20
2.3	<i>Convolutional Neural Networks</i>.....	21
2.3.1	Redes CNN pré-treinadas	24
2.3.2	VGG-19	25
2.3.3	ResNet-50	27
2.4	Análise de Componentes Principais.....	30
2.5	Métricas de avaliação	30
2.5.1	Matriz de Confusão multiclasse	30
2.5.2	Acurácia.....	32
2.5.3	Sensibilidade.....	33
2.5.4	Precisão.....	33
2.5.5	F1-score	33
2.5.6	Médias Macros	33
3	METODOLOGIA	35
3.1	Base de dados	35
3.2	Pré-processamento e aumento de dados.....	36
3.3	Rede CNN.....	39
3.4	Extração de características	41
3.5	Modelo SVM	42
3.6	Modelo XGBoost.....	42
3.7	Recursos Computacionais.....	42
4	RESULTADOS.....	44

4.1 Resultados da VGG-19.....	44
4.2 Resultados da ResNet-50.....	46
4.3 Resultados da ResNet-50 + SVM.....	48
4.4 Resultados da ResNet-50 + XGBoost.....	49
4.5 Comparação dos resultados.....	52
5 CONCLUSÕES E TRABALHOS FUTUROS	54
5.1 Conclusões	54
5.2 Trabalhos futuros	54
REFERÊNCIAS BIBLIOGRÁFICAS	56

1 INTRODUÇÃO

O câncer de mama é um dos dois tipos mais comuns de cânceres sendo mais diagnosticado em mulheres acima dos 40 anos, ainda assim, há relatos de mulheres que foram diagnosticadas antes dos 30 e este tipo de câncer também pode ser detectado em homens, mesmo que com uma frequência consideravelmente menor. De acordo com a organização mundial da saúde (do inglês *World Health Organization* – WHO), em 2020 2,3 milhões de mulheres foram diagnosticadas com a doença. Resultando em 685 mil óbitos, o que representa uma taxa de mortalidade de quase 30%, proporção muito grande em relação ao total de casos (WORLD HEALTH ORGANIZATION, 2021). No início da doença é comum a ausência de sintomas, que normalmente são notados quando o tumor já está bastante avançado, inclusive, este é um dos principais motivos da alta mortalidade em áreas rurais, pois nestes casos a detecção acaba sendo tardia e com isso a chance de sucesso no tratamento diminui (THANI; KASBE, 2020).

O diagnóstico precoce para essa doença é fundamental, sendo o método mais comum de identificação do tumor é por meio de mamografias. Felizmente, o assunto tem cada vez mais atraído o interesse de pesquisadores dos campos de visão computacional e aprendizado de máquina. Proporcionando avanços na identificação desta enfermidade, já que com a ajuda de métodos computacionais de inteligência artificial o médico possui mais ferramentas disponíveis na hora de identificar e tratar a doença.

Por meio da utilização de imagens, tanto de mamografias quanto de tecidos das mamas, modelos de aprendizado de máquina estão sendo utilizados para a detecção e classificação dos diferentes tipos de câncer, tornando possível diferenciar as imagens com o tecido normal, tumor benigno, câncer não invasivo e câncer invasivo – quando o tumor se espalha para outras áreas.

Diante disso, este trabalho busca aplicar métodos computacionais para a caracterização e classificação de mamografias digitais quanto à presença de câncer com a utilização de aprendizado de máquina (do inglês, *Machine Learning* – ML) e aprendizado profundo (do inglês, *Deep Learning* – DL), logo, espera-se a elaboração de um modelo que seja capaz de classificar os diferentes tipos de tecidos por meio de informações extraídas de imagens,

servindo como forma complementar para o diagnóstico da doença e assim auxiliando os profissionais de saúde na hora da tomada de decisão.

1.1 Justificativas

A detecção precoce da doença é uma das ferramentas mais eficazes no tratamento da doença, para isso, a conscientização sobre o tema é muito importante para que as pessoas saibam como observar os primeiros sintomas e então possam se dirigir a um profissional da saúde para acompanhar qualquer irregularidade (KIM; ARAUJO; TSAI et al., 2021).

O avanço da tecnologia vem ajudando cada vez mais a área da saúde na detecção e tratamento de doenças por meio de métodos de aprendizado de máquina (do inglês, *machine learning*). Tais métodos computacionais são capazes de aprender com base em exemplos disponibilizados a eles. Os modelos podem ser supervisionados, onde exemplos rotulados de dados são apresentados ao modelo para que então a saída do modelo seja comparada ao valor esperado, ou podem ser modelos não supervisionados que, ao contrário dos modelos anteriores, não possui um valor de referência para comparar a saída, com isso os modelos mapeiam as características intrínsecas dos dados de acordo com as suas semelhanças e diferenças (MURPHY, 2012).

Com o objetivo de auxiliar a tomada de decisão do profissional da saúde a respeito da avaliação sobre a possível presença ou não da doença, diversos métodos de aprendizado de máquina, principalmente modelos de aprendizado profundo (do inglês, *deep learning*) e visão computacional, foram desenvolvidos para auxiliar neste tipo de decisão. É muito importante se ter um alto índice de acerto, pois classificações incorretas podem trazer grandes impactos na vida dos pacientes causando complicações desnecessárias. Por exemplo, um paciente pode ser submetido a um tratamento extremamente rigoroso para tratar uma doença inexistente ou uma doença muito grave não ser tratada da forma correta trazendo risco de morte.

A maior capacidade de processamento dos computadores trouxe melhorias para o campo de classificação de imagens em virtude de possibilitar a execução de modelos complexos. Graças ao processamento em paralelo, que ganhou força com o avanço das unidades de processamento gráfico (do inglês, *graphic processing unit* – GPU), métodos de aprendizado baseado na atividade cerebral do ser humano puderam ser implementados de forma mais

complexa para a identificação e classificação de objetos (LECUN et al., 2018). A partir desse aumento da capacidade de processamento, as redes se tornaram capazes de processar um grande conjunto de dados. Tornando possível executar redes extremamente complexas que são capazes de mapear um conjunto de dados de grandes proporções, com um grau de detalhamento e precisão maior do que técnicas mais simples de aprendizado de máquina. Isso é possível pela capacidade que a rede neural profunda tem de extrair características dos dados com o auxílio do *backpropagation*, método utilizado que possibilita ajustes e maior capacidade de generalização das redes (BROWNLEE, 2019).

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho tem como propósito comparar diferentes modelos de classificação para a detecção de câncer de mama em mamografias. Para isso, será utilizada a base de dados disponibilizada na 15ª Conferência Internacional de Análise e Reconhecimento de Imagens (ICIAR, 2018). O conjunto contém imagens de microscopia celular separadas em 4 classes: células normais, células com tumor benigno, células cancerígenas com tumor não invasivo e células cancerígenas com tumor invasivo.

1.2.2 Objetivos específicos

1. Estudar e implementar os métodos de classificação que serão utilizados, que serão redes neurais convolucionais (do inglês, *Convolutional Neural Network* – CNN), máquinas de vetores de suporte (do inglês, *Support Vector Machine* – SVM) e uma técnica baseada em árvore de decisão chamada *Extreme Gradient Boosting* (XGBoost);
2. Extrair características do conjunto de imagens usando CNN;
3. Comparar os desempenhos dos três métodos de classificação.

1.3 Organização do Trabalho

- Capítulo 1 – Introdução: Contextualização do problema estudado apresentando os pontos mais relevantes e a proposta do projeto;
- Capítulo 2 – Referencial Teórico: Detalha as técnicas e métodos que serão empregados no desenvolvimento do trabalho explicando os modelos que serão utilizados e formas de avaliação de desempenho;
- Capítulo 3 – Metodologia: Detalha a metodologia implementada no desenvolvimento do projeto, descrevendo o banco de dados que foi utilizado, a forma com que os dados foram processados e como foi feita a extração de características que foram utilizadas nas demais técnicas de aprendizado de máquina;
- Capítulo 4 – Resultados: Os resultados obtidos por meio da metodologia apresentada no capítulo anterior são apresentados neste capítulo;
- Capítulo 5 – Conclusões e Trabalho Futuros: Neste último capítulo são apresentadas algumas considerações que foram obtidas após a análise dos resultados obtidos no desenvolvimento do trabalho, assim como, sugestões de trabalhos futuros para obtenção de resultados melhores.

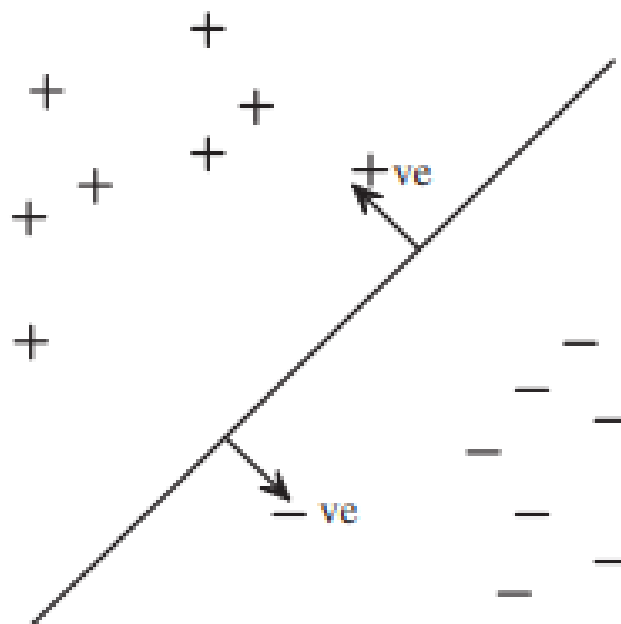
2 REFERENCIAL TEÓRICO

Este capítulo tem como objetivo introduzir conceitos que foram utilizados para o desenvolvimento do trabalho, nele serão descritas as técnicas utilizadas, começando pelas máquinas de vetores de suporte, métodos baseados em árvores de decisão, redes neurais convolucionais e, por fim, métricas de avaliação de desempenho dos algoritmos.

2.1 *Support Vector Machines*

O classificador máquina de vetores de suporte (do inglês, *Support Vector Machine* – SVM) é uma técnica muito utilizada na identificação de padrões em tarefas de classificação binária (duas classes), transformando os dados de entrada por meio de elevação de dimensionalidade (CORTES; VAPNIK, 1995). A separação entre duas classes em duas dimensões é feita por uma reta que separa os dados por meio da disposição deles em relação à reta. Para os casos com mais de duas dimensões a separação é feita por meio de hiperplanos (THEODORIDIS, 2009). A Figura 1 ilustra um exemplo de duas classes separadas por um hiperplano, no caso, uma reta.

Figura 1 – Exemplo de uma reta separando duas classes



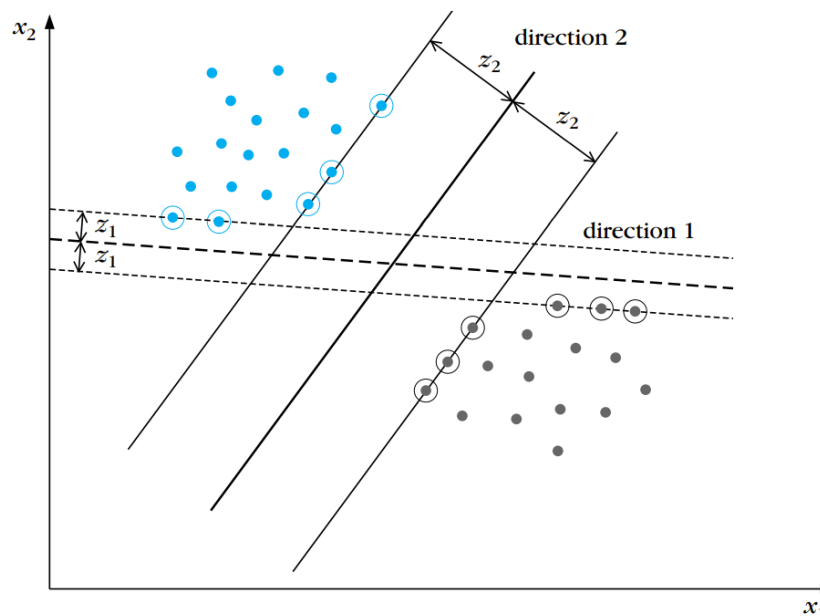
Fonte: Campbell (2011)

Este modelo é uma técnica de aprendizado supervisionado, ou seja, o aprendizado do método ocorre com dados rotulados durante uma etapa de treinamento e, só após o algoritmo aprender o padrão de cada classe, são apresentados novos dados para que seja possível o modelo inferir qual é classe destes novos dados (HEYMAN, 2019).

Para tanto, não basta o modelo simplesmente ser capaz de separar os dados, mas também é importante que ele tenha um bom desempenho na generalização de dados, pois assim será apto de ter um bom desempenho quando submetido a um novo conjunto de dados ainda não conhecidos pelo modelo. Tal habilidade de generalizar é muito importante para evitar o chamado *overfitting*, que acontece quando o algoritmo não consegue manter o desempenho visto no treinamento quando submetido a novos elementos (LECUN; BENGIO; HINTON, 2018).

Com o intuito de melhor separar os dados e conseguir manter uma boa distância entre as classes, a margem de separação entre as classes é obtida por meio de uma função que tem como principal objetivo minimizar o erro nos dados de treino para assim maximizar a distância entre uma classe e outra (THEODORIDIS, 2009). A Figura 2 demonstra um exemplo de margens separando duas classes.

Figura 2 – Margens de um Hiperplano



Fonte: Theodoridis (2009)

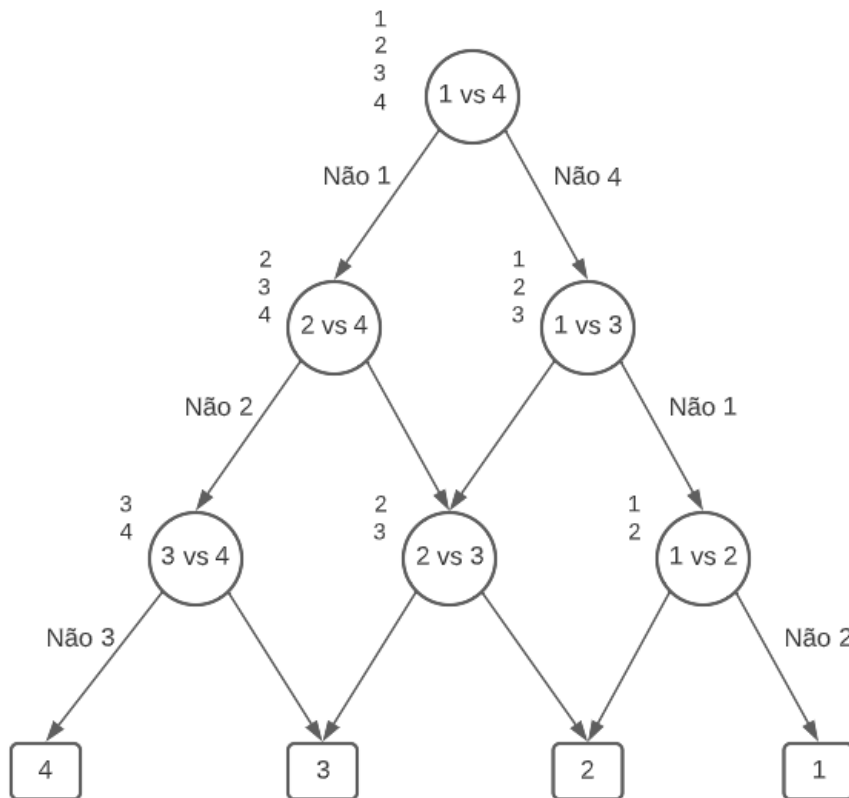
Nos casos em que há mais de duas classes, as SVM utilizam conjuntos de treinamento compostos por pares (x_i, y_i) , tal que $y_i \in \{1, \dots, k\}$ para $k > 2$, sendo k o número de classes. Os métodos mais comuns de se solucionar estes problemas são por meio da decomposição em k problemas binários, pelas técnicas “Um-Contra-Todos” e “Todos-Contra-Todos” (BISHOP, 2006).

O método “Um-Contra-Todos” se baseia na construção de k classificadores binários onde cada um se torna responsável pela classificação de apenas uma classe e a escolha do resultado final se dá por meio daquele que obteve o maior valor conforme demonstrado na Equação 1.

$$f(x) = \arg \max_{1 \leq i \leq k} (f_i(x)) \quad (1)$$

O método “Todos-Contra-Todos”, para um problema com k classes, faz a comparação em pares criando $k(k-1)/2$ máquinas de vetores de suporte que classificam por meio de votações com o objetivo de eliminar as possibilidades até que reste apenas uma. A Figura 3 demonstra o método para um problema com 4 classes.

Figura 3 – Método Todos-Contra-Todos para 4 classes.

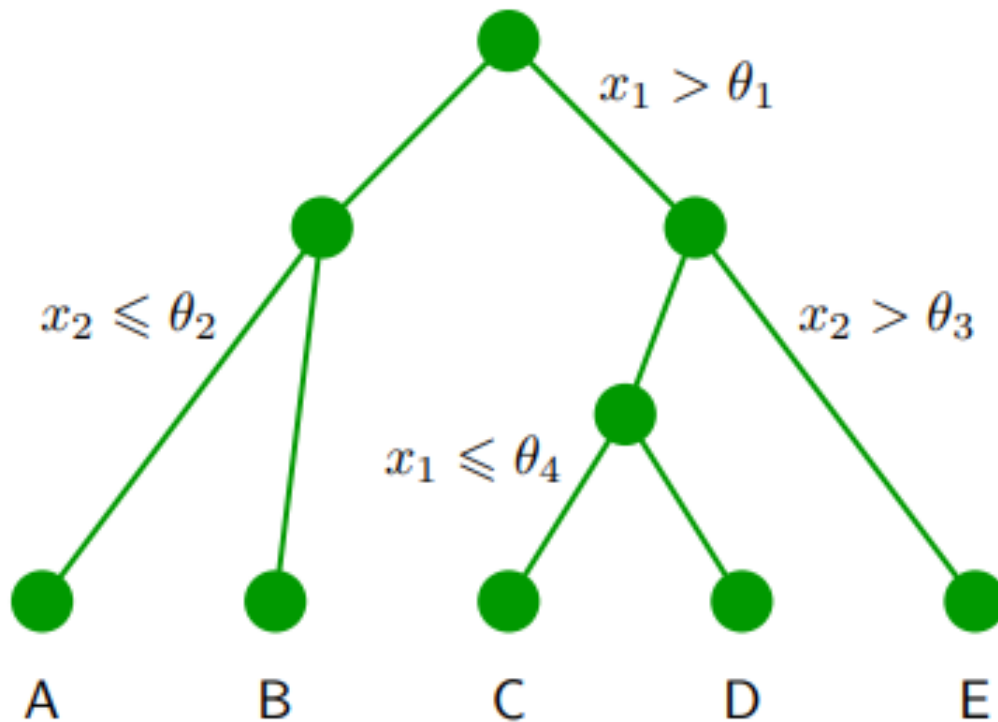


Fonte: Produção do próprio autor.

2.2 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) é uma técnica baseada em árvores de decisão que são algoritmos simples e que apresentam bons desempenhos em classificar informações. A árvore se baseia na técnica de “dividir para conquistar”, que decompõe um problema maior em subproblemas com características simplificadas. Em seguida, a técnica é reaplicada no subproblema de forma recursiva até o conjunto de dados consiga ser explicado de forma satisfatória (BREIMAN et al, 1984). A Figura 4 traz um exemplo de árvore de decisão.

Figura 4 – Exemplo de árvore de decisão



Fonte: Bishop (2006).

O XGBoost se baseia no modelo de *gradient boosting model* (GBM) proposto por Friedman em 1999, que otimiza o aprendizado por meio do gradiente descendente (FRIEDMAN, 2000). O objetivo principal é criar diversas árvores de decisões, com o objetivo de reduzir o erro residual destes submodelos para que o modelo seja capaz de evoluir no âmbito de generalização dos dados. Os parâmetros da árvore principal são selecionados a partir da combinação das subdivisões que apresentam os menores erros residuais (CHEN; GUESTIN, 2016).

2.3 Convolutional Neural Networks

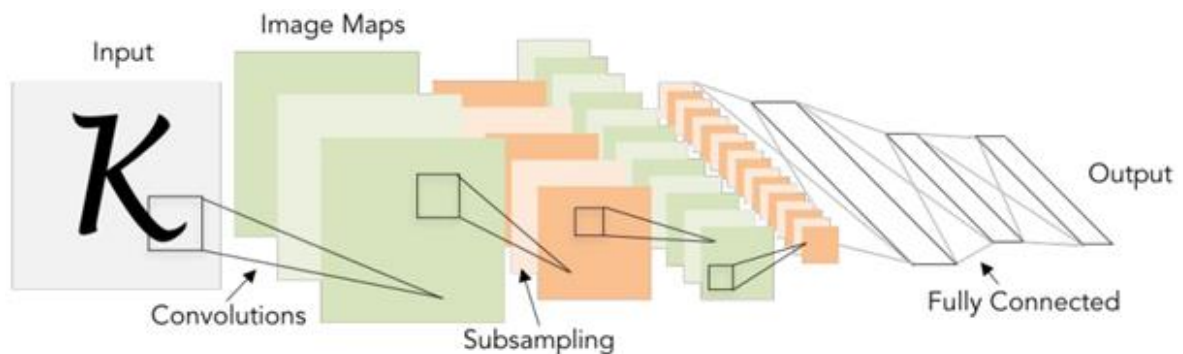
Estas redes são o estado da arte no quesito de processamento e análise em imagens digitais $I(i,j)$, que por definição possuem M linhas e N colunas, onde cada componente da matriz é denominada de pixel. Além disso, para as imagens coloridas existem 3 canais que representam a intensidade das cores vermelho (*red*), verde (*green*) e azul (*blue*), o que forma o espaço de cor RGB.

Tais modelos são baseadas em convolução, que é um cálculo matemático baseado na superposição de duas funções para gerar uma terceira função que é a soma das demais ao longo da região a ser analisada. Uma convolução 2D entre uma imagem digital I e uma máscara K , também conhecida como *kernel*, é apresentada na Equação 2. A máscara possui dimensão $M \times N$ e a imagem I possui dimensão maior do que a máscara. O objetivo do *kernel* é alterar características da imagem (seja suavizando, realçando bordas, entre outras.) e obter uma imagem de saída S (GOODFELLOW; BENGIO; COURVILLE, 2016).

$$S(i,j) = K(i,j) \star I(i,j) = \sum_{m \in M} \sum_{n \in N} K(m,n)I(i-m,j-n) \quad (2)$$

Estas redes estão entre os modelos de aprendizado profundo mais utilizados na atualidade, tendo uma grande gama de aplicações utilizando este tipo de algoritmo. No geral, uma rede CNN possui três camadas principais, sendo elas: Camada Convolutiva, Camada de *pooling* e Camada Totalmente Conectada (BENHAMMOU et al., 2018). A Figura 5 demonstra um exemplo de rede neural convolucional.

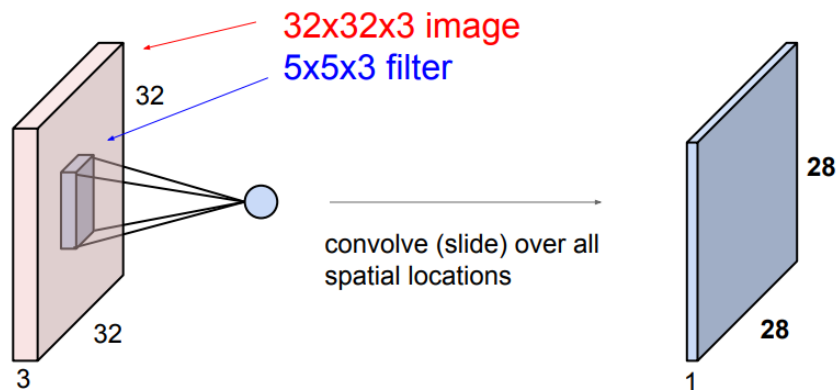
Figura 5 – Exemplo de uma CNN.



Fonte: LECUN et al. (2015).

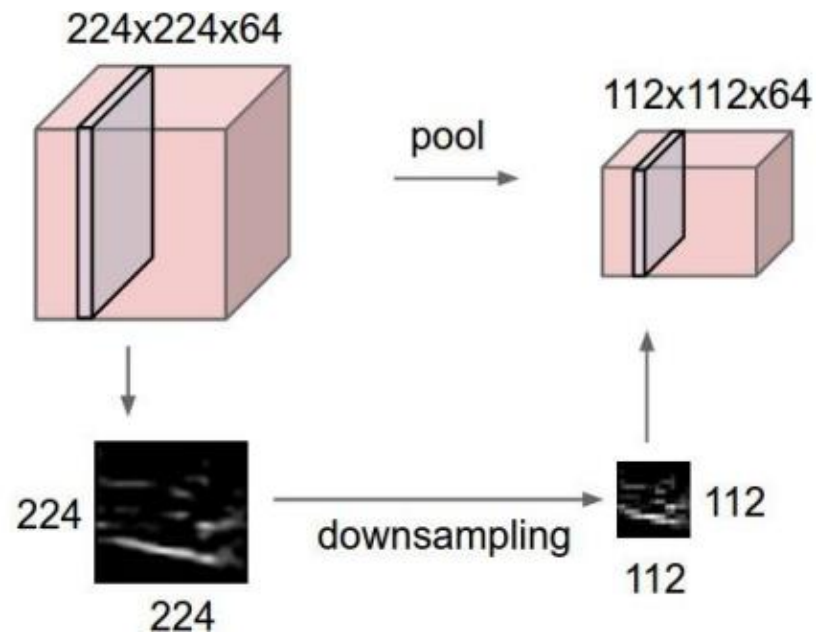
A camada convolucional é responsável por detectar padrões nos dados de entrada. Esta camada é composta por vários kernels de convolução, sendo as saídas organizadas em planos, também conhecidos como mapa de características. As unidades no mapa de características são os resultados da soma ponderada de sub-regiões da imagem ponderadas pelos pesos dos kernels. A identificação dos padrões é feita por meio de funções de ativação, que aplicadas aos resultados da convolução, podendo ser do tipo sigmoid, tangente hiperbólico e também ReLu, conforme demonstrado na Figura 6 (GOODFELLOW; BENGIO; COURVILLE, 2016) (BISHOP; NASRABADI, 2006).

Figura 6 – Camada de convolução pra obtenção de características



Fonte: KARPATY, A. et al. (2017)

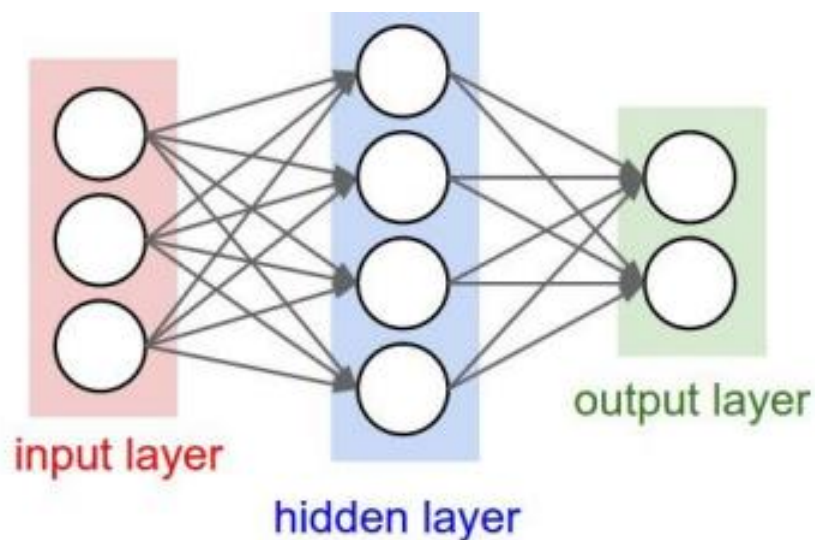
Já a camada de *pooling* é responsável pela redução de dimensionalidade das informações, ou seja, simplificar os dados com o objetivo de se destacar as características mais importantes em detrimento daquilo que não possui tanta relevância para a segmentação de característica. Esta camada funciona como uma operação de *downsampling* de sua entrada, pois visa reduzir as dimensões espaciais da imagem com base em operações matemáticas, como a média dos parâmetros vizinhos, *average pooling*, ou métodos que destacam os maiores valores da região analisada, também conhecido como *max-pooling*. Tal etapa se mostra eficaz na redução do custo computacional dos modelos (GOODFELLOW; BENGIO; COURVILLE, 2016) (KARPATY *et al*, 2017). A Figura 7 demonstra a operação de *pooling*.

Figura 7 – Exemplo de *pooling*

Fonte: KARPATY, A. et al. (2017)

A camada totalmente conectada (do inglês *fully connected* – FC), conforme demonstrada na Figura 8, tem como função realizar a classificação da imagem de entrada a partir das características extraídas nas camadas convolucionais da rede (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 8 – Camada totalmente conectada



Fonte: KARPATY, A. et al. (2017)

Além disso, a rede permite extrair características e utilizar as informações geradas pelos filtros das camadas convolucionais como entrada para outros classificadores, como as máquinas de vetores de suporte. Para isso, é utilizada uma camada *flatten* que transforma os dados em um vetor (ARAÚJO et al., 2017).

2.3.1 Redes CNN pré-treinadas

Treinar uma rede neural convolucional é uma tarefa extremamente complicada e demorada pois requer um conjunto grande de dados com milhares de amostras. Com isso, é comum realizar algumas adaptações em redes que foram previamente treinadas para objetivos que são similares ao que será realizado. Tal técnica é chamada de *transfer learning* (PAN; KWOK; YANG, 2008).

A metodologia de aplicação do *transfer learning* é baseada em reutilizar a estrutura de uma rede que foi treinada para uma tarefa específica em problemas com características parecidas. Para as redes utilizadas na classificação de imagens, as primeiras camadas são preservadas já que são responsáveis pela identificação de características mais simples, como a identificação de padrões de bordas e texturas (ZEILER, M.; FERGUS, 2014). Já as camadas de saída são responsáveis por classificar os dados com base no conjunto apresentado, e elas costumam ser treinadas com os dados da tarefa que se pretende realizar. As redes utilizadas para este projeto foram previamente treinadas com a base de dados *ImageNet*, que é um dos conjuntos mais utilizados no treinamento de modelos de *deep learning* (RUSSAKOVSKY et al, 2009).

Como o número e tipos de classes da *ImageNet* são diferentes das utilizadas neste trabalho, foi necessária a remoção da última camada totalmente conectada da rede pré-treinada, sendo substituída por uma nova camada que será treinada com a base de dados deste trabalho. Tal tarefa é conhecida como *fine tuning* que consiste em reutilizar as características intermediárias da rede de forma a se adaptar ao novo conjunto de informações (ALKHALEEF AH; WU, 2018).

2.3.2 VGG-19

A VGG foi uma rede criada por Karen Simonyan e Andrew Zisserman e foi desenvolvida para o desafio ILSVRC - *ImageNet Large Scale Visual Recognition Challenge* do ano de 2014 onde obtiveram a primeira colocação com uma taxa de erro de apenas 25,30% (SIMONYAN; ZISSERMAN, 2015) (RUSSAKOVSKY et al, 2009). Atualmente é uma das mais utilizadas para problemas de reconhecimento e classificação de imagens por conta de sua estrutura simplificada, quando comparada com outras redes convolucionais, além de apresentar bons valores de acurácia em baixos períodos de treinamento. Na Tabela 1 é detalhada a estrutura da rede VGG-19 que é utilizada neste trabalho.

Tabela 1 – Estruturas da rede VGG-19

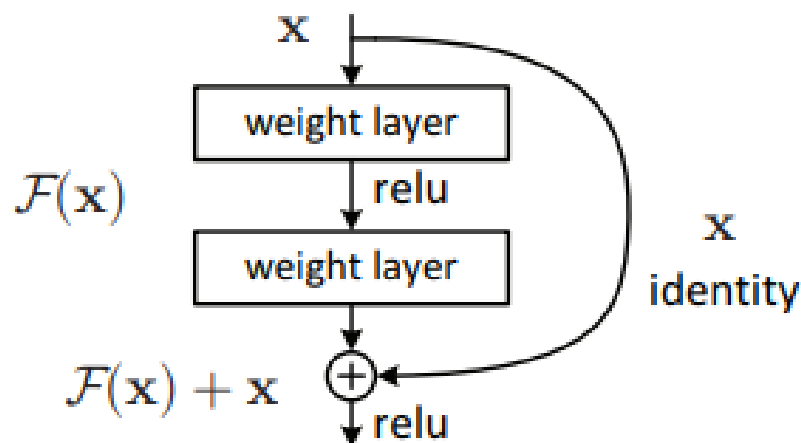
Camada	Filtro
Entrada (Imagem RGB)	224x224x3
Convolutacional	224x224x64
Convolutacional	224x224x64
<i>Max pooling</i>	
Convolutacional	112x112x128
Convolutacional	112x112x128
<i>Max pooling</i>	
Convolutacional	56x56x256
Convolutacional	56x56x256
Convolutacional	56x56x256
Convolutacional	56x56x256
<i>Max pooling</i>	
Convolutacional	28x28x512
Convolutacional	28x28x512
Convolutacional	28x28x512
Convolutacional	28x28x512
<i>Max pooling</i>	
Convolutacional	14x14x512
Convolutacional	14x14x512
Convolutacional	14x14x512
Convolutacional	14x14x512
<i>Max pooling</i>	
Totalmente conectada	1x1x4096
Totalmente conectada	1x1x4096
Totalmente conectada	1x1x1000

Fonte: Produção do próprio autor.

2.3.3 ResNet-50

O nome da rede é a abreviação de *Residual Network* e foi proposta por engenheiros da *Microsoft*. Podendo a estrutura variar conforme o número de camadas, onde as mais simples possuem 18 e as mais complexas 152 camadas, como na ResNet-152, que possui 8 vezes mais camadas que a rede VGG19 possibilitando maior detalhamento na extração de características. Porém, o número elevado de camadas aumenta a chance de saturar o desempenho do modelo e, conseqüentemente, reduzir a capacidade de generalizar as informações em virtude do gradiente das operações entre as camadas tender a reduzir a zero, fato conhecido como *vanishing gradients* (He et al. 2016). Para resolver tal problema da rede foram adicionados atalhos entre as camadas (do inglês, *shortcut connections*) com o objetivo de que o gradiente possa ser transmitido entre camadas de forma que não exista perda de desempenho entre uma camada e outra. A Figura 9 ilustra isto.

Figura 9 – Atalhos entre camadas.



Fonte: HE et al. (2016).

Na Tabela 2 estão dispostas algumas configurações para a ResNet começando pelas redes mais simples com 18 e 34 camadas e indo até redes mais complexas com 101 e 152 camadas. Com o objetivo de otimizar o custo computacional, nas duas primeiras linhas são aplicadas técnicas de redução espacial, conhecidas como *stride*, na tradução livre, passos. Esta técnica tem como objetivo reduzir o número de informações pela metade, por meio de uma função de ativação que é calculada a cada dimensão espacial.

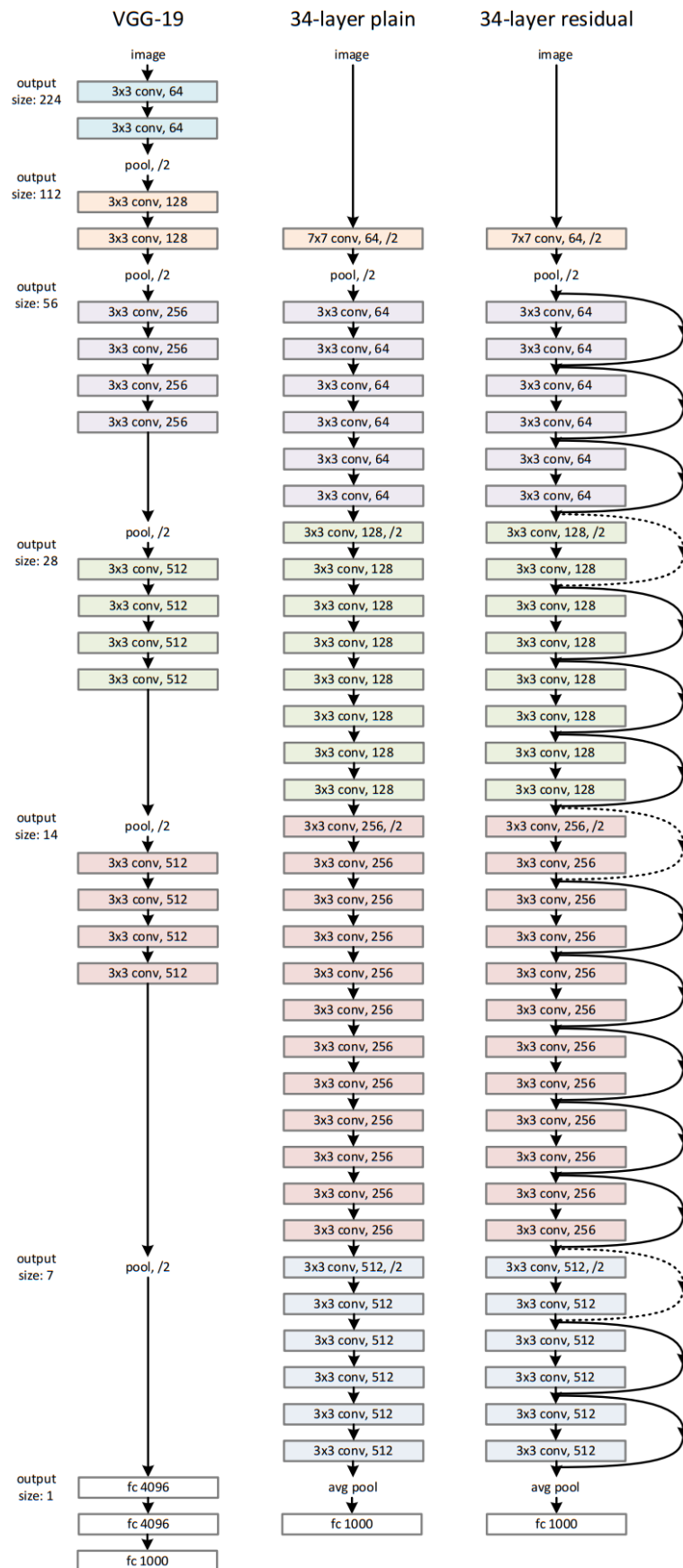
Tabela 2 – Configurações da rede ResNet

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10 ⁹	3.6×10 ⁹	3.8×10 ⁹	7.6×10 ⁹	11.3×10 ⁹

Fonte: He. (2016).

De forma complementar, a Figura 10 mostra a comparação entre a rede VGG-19 e a ResNet-34. Na parte a esquerda da figura demonstra a rede VGG-19, enquanto a parte central da figura demonstra uma rede genérica com 34 camadas. Já a imagem mais à direita, demonstra como as conexões de salto foram introduzidas, com o objetivo de otimizar o desempenho da rede, corrigindo assim o problema do *vanishing gradient* (HE *et al*, 2016).

Figura 10 – Comparação entre VGG-19 e ResNet-34.



Fonte: He. (2016).

2.4 Análise de Componentes Principais

A Análise de Componentes Principais (do inglês, *Principal Components Analysis* – PCA) tem como objetivo reduzir a quantidade de informações necessárias que sejam capazes de se explicar certo conjunto de dados. Ou seja, transformar uma certa quantidade de características iniciais em um novo grupo com dimensão menor definida pela quantidade de componentes principais usadas para representar este novo conjunto de dados. Tais componentes possuem a propriedade de serem linearmente independentes entre si e poder calcular a variabilidade dos dados na direção de cada componente principal. A redução de dimensionalidade dos dados originais é possível descartando as componentes principais associadas às direções com os menores valores de dispersão dos dados (GOODFELLOW; BENGIO; COURVILLE, 2016).

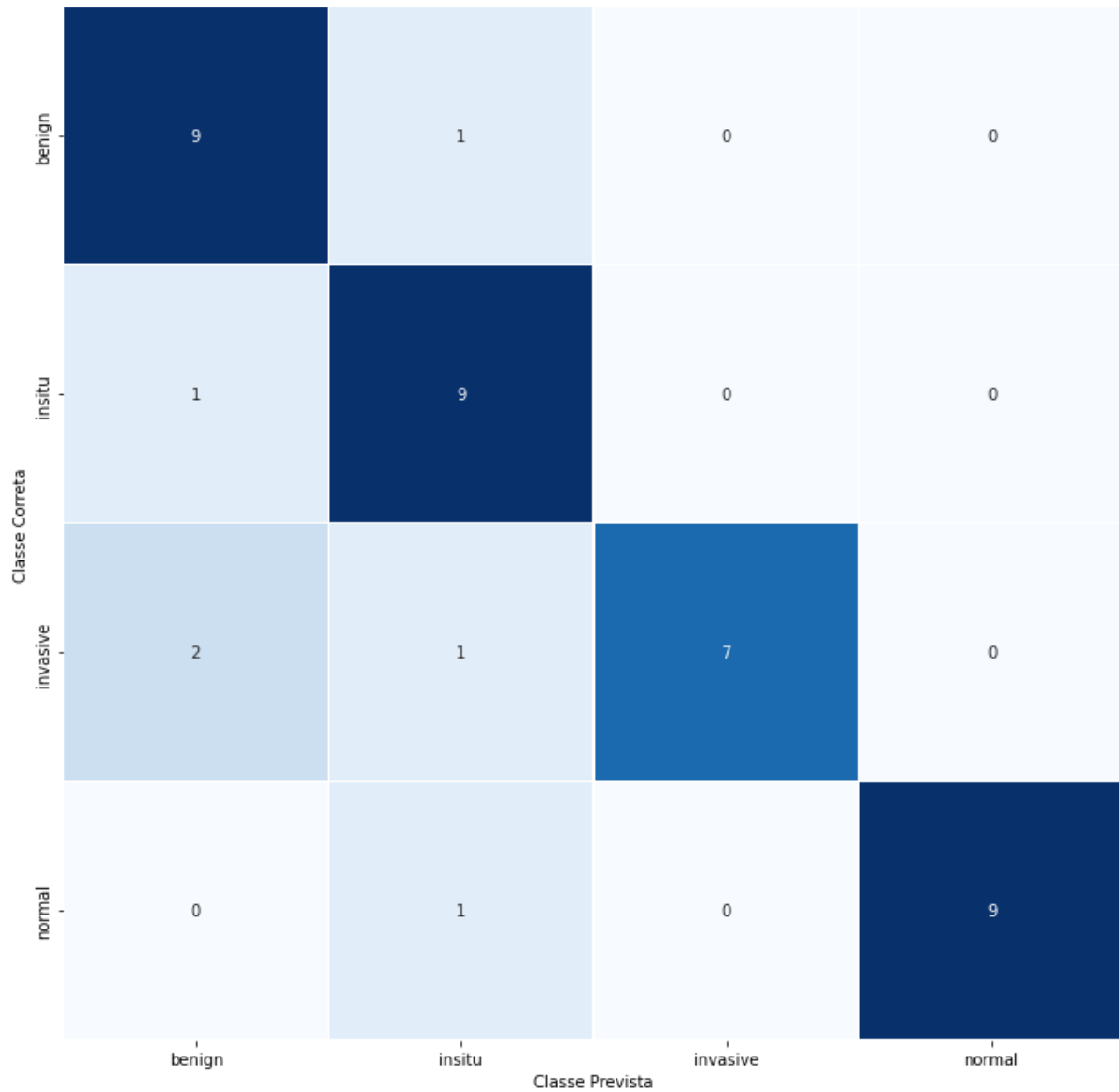
Esta técnica é muito utilizada em problemas com um grande número de informações, para que de uma forma otimizada seja possível trabalhar com um número menor de dados, porém ainda com muitas informações dos dados originais.

2.5 Métricas de avaliação

2.5.1 Matriz de Confusão multiclasse

Uma das formas mais utilizadas para análise de desempenho de modelos de classificação é a matriz de confusão, sendo possível estimar quão bem o algoritmo é capaz de prever a classe correta do dado de entrada. Diferentemente da matriz de confusão binária onde só há duas opções de classes, neste tipo há mais opções de classificação e o resultado final é uma matriz $N \times N$, onde N é o número de classes (MARKOULIDAKIS, 2021). A Figura 11 demonstra um exemplo de matriz de confusão com 4 classes distintas.

Figura 11 – Exemplo de matriz de confusão com 4 classes.



Fonte: Produção do próprio autor.

Para os problemas com mais de duas classes, a análise das métricas de avaliação do modelo deve primeiro ser feita para cada classe, conforme demonstrado na Figura 12 que analisa os parâmetros para a classe C_2 .

Figura 12 – Análise de métricas para a Classe C_2

		Classe Prevista			
		C_1	C_2	...	C_N
Classe Real	C_1	$C_{1,1}$	FP	...	$C_{1,N}$
	C_2	FN	TP	...	FN
	...	...	...	...	...
	C_N	$C_{N,1}$	FP	...	$C_{N,N}$

Fonte: MARKOULIDAKIS (2021).

Conforme apresentado na Figura 12, os valores que estão na diagonal principal são chamados de verdadeiro positivos (do inglês, *True Positive* – *TP*) e corresponde àquilo que foi previsto de forma correta, enquanto os demais valores da segunda coluna representam falsos positivos (do inglês, *False Positive* – *FP*), isto é, dados que foram classificados erroneamente na classe C_2 (a classe analisada). Já os valores da segunda linha que não fazem parte da segunda coluna são chamados de falsos negativos (do inglês, *False Negative* – *FN*), isto é, dados da classe C_2 que foram classificados em outras classes.

2.5.2 Acurácia

A acurácia do modelo se dá pelo número de casos em que a classificação foi correta, ou seja, o somatório da diagonal principal dividido pelo total de amostras, conforme a Equação 3.

$$Acurácia\ do\ Modelo = \frac{\sum_{i=1}^N TP(C_i)}{n^o\ Total\ de\ amostras} \times 100\% \quad (3)$$

Importante ressaltar que não basta olhar apenas para a acurácia quando se deseja analisar o desempenho do classificador e para isso outras métricas devem ser levadas em consideração.

2.5.3 Sensibilidade

A sensibilidade para cada classe, também conhecida como *recall*, é a taxa de verdadeiros positivos (do inglês, *True Positive Rate – TPR*), isto é, a razão entre os dados classificados corretamente como positivos em relação ao total de dados positivos nos dados avaliados, como mostrada na Equação 4.

$$TPR \text{ da Classe}(C_i) = \frac{TP(C_i)}{TP(C_i) + FN(C_i)} \times 100\% \quad (4)$$

2.5.4 Precisão

Para cada classe, a precisão, também conhecida como *Positive Predictive Value – PPV*, é obtida pela relação entre os dados classificados corretamente como positivos e o número de objetos que foram classificados como positivos pelo modelo, como apresentada na Equação 5.

$$PPV \text{ da Classe}(C_i) = \frac{TP(C_i)}{TP(C_i) + FP(C_i)} \times 100\% \quad (5)$$

2.5.5 F1-score

F_1 -score é obtido pela média harmônica entre a precisão e a sensibilidade do modelo, como apresentado na Equação 6.

$$F_1 - score \text{ da Classe}(C_i) = 2 * \frac{PPV * TPR}{PPV + TPR} \times 100\% \quad (6)$$

2.5.6 Médias Macros

Para analisar o desempenho geral de um modelo com N classes, calcula-se as médias macro dos indicadores conforme as Equações 7, 8 e 9.

$$TPR (Macro) = \frac{1}{N} * \sum_{i=1}^N TPR(C_i) \quad (7)$$

$$PPV (Macro) = \frac{1}{N} * \sum_{I=1}^N PPV(C_i) \quad (8)$$

$$F_1 - score (Macro) = 2 * \frac{TPR (Macro) * TPR (Macro)}{TPR (Macro) + TPR (Macro)} \quad (9)$$

3 METODOLOGIA

A metodologia proposta busca realizar uma abordagem analítica, comparando técnicas diferentes de aprendizado de máquina de classificação para a análise e comparação dos resultados quando submetidos a uma base de dados.

Será utilizada uma rede CNN para classificação e extração de características em imagens digitalizadas que serão utilizadas para treinar outros dois modelos de classificação. Além disso, serão feitos experimentos com o objetivo de maximizar o desempenho dos classificadores, buscando assim otimizar e trazer maior assertividade para os modelos. Por fim, os resultados dos modelos serão comparados.

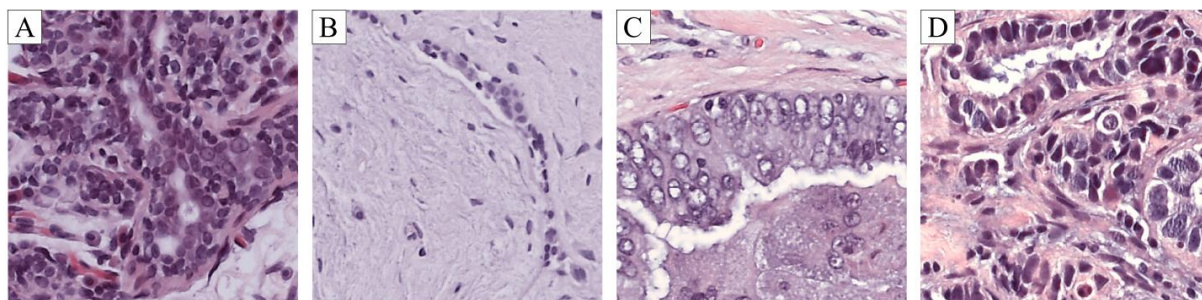
3.1 Base de dados

Com o propósito de ajudar a comunidade médica com o diagnóstico precoce de câncer de mama, imagens microscópicas de tecido da mama foram obtidas de diversas pacientes, tanto aquelas que possuíam câncer como aquelas que estavam livres da doença. A base de dados foi disponibilizada de forma gratuita, disponível no site da ICIAR 2018, para o *Grand Challenge on Breast Cancer Histology Images* – BACH (ICIAR, 2018).

O principal objetivo do desafio foi obter formas de classificação automática de imagens de microscopia celular para a detecção de câncer de mama. Ao todo, houve 677 registros para a participação do evento, porém, aconteceram apenas 64 envios para avaliação, o que obteve o melhor desempenho apresentou 87% de acurácia (ARESTA; ARAÚJO; KWOK; et al., 2019).

O conjunto de dados possui ao todo 400 imagens rotuladas em quatro classes divididas de forma igualitária entre 4 classes, logo, cada classe possui 100 exemplares de alta resolução com 2048×1536 pixels. As imagens foram rotuladas conforme o tipo de patologia podendo ser: normal, benigno (Benign), câncer não invasivo (In situ carcinoma) e câncer invasivo (Invasive carcinoma) (KONÉ; BOULMANE, 2018). A Figura 13 demonstra exemplos de imagens de cada classe.

Figura 13 – Exemplo das imagens microscópicas. **A** – Normal, **B** – Benigno, **C** – Câncer não invasivo, **D** – Câncer invasivo



Fonte: Araújo *et al.* (2020)

Para a coloração das imagens foi utilizada a técnica Hematoxilina – Eosina (H&E), que atualmente é uma das técnicas mais empregadas na visualização de tecidos e estruturas celulares. O núcleo da célula e partes que contêm RNA, por conter características ácidas ao entrar em contato com a Hematoxilina, apresentam uma coloração mais escura. Por outro lado, por possuir características básicas, o citoplasma apresenta tons rosados quando expostos a Eosina.

3.2 Pré-processamento e aumento de dados

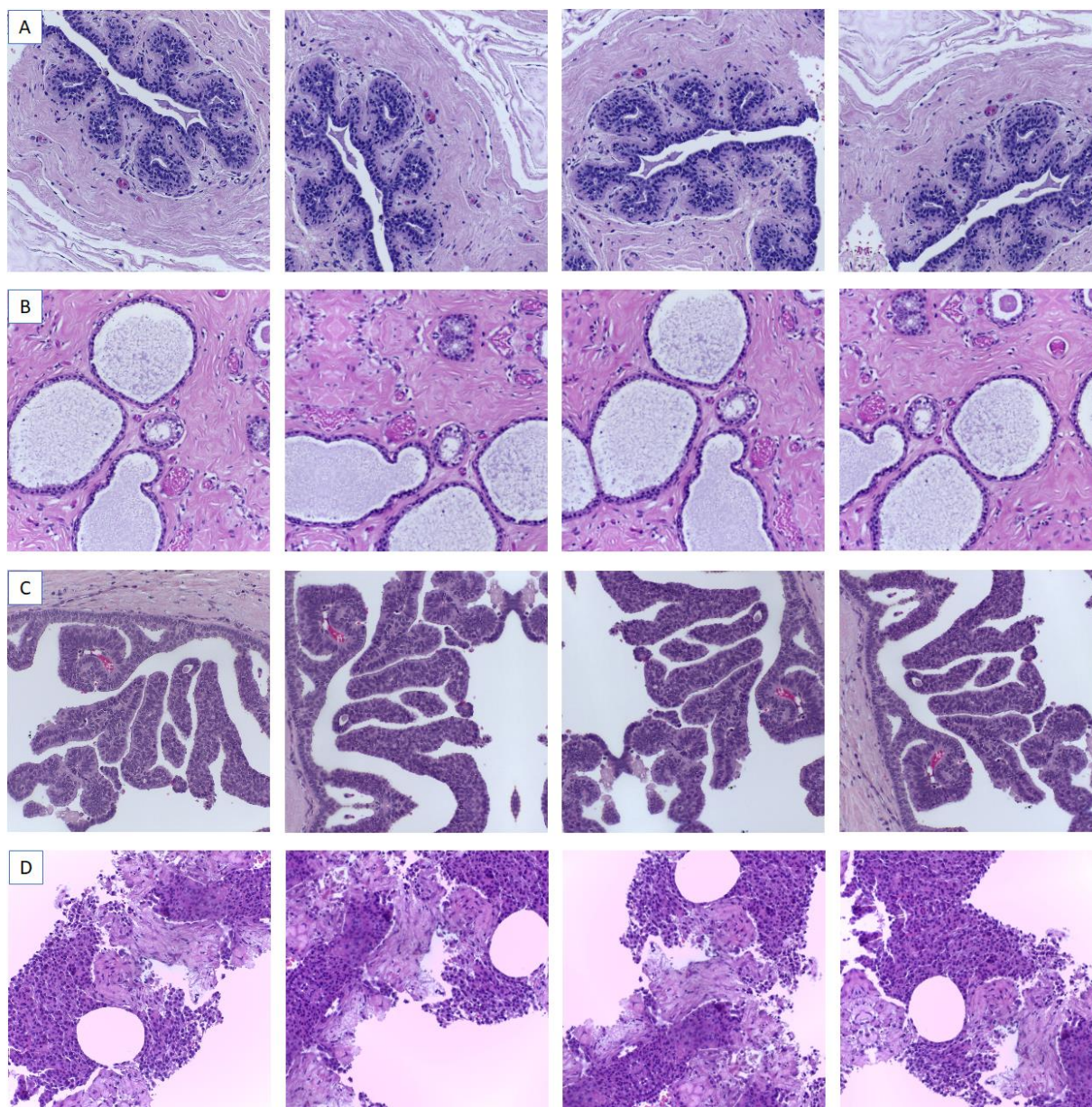
Os dados foram salvos em pastas separadas pelos rótulos de treino, validação e teste, o primeiro conjunto tem 300 imagens (75%), o segundo possui 60 imagens (15%) e o terceiro tem 40 imagens (10%). A divisão foi feita por meio de pastas que foram divididas conforme a etapa em questão *train*, *val* e *test* e o rótulo dos dados podendo ser: *benign*, *in situ*, *invasive* e *normal*.

As imagens originais eram de alta resolução com aproximadamente 13 MB de tamanho e resolução de 2048×1536 pixels. Por conta da elevada dimensionalidade e tamanho de cada exemplar, foi necessário modificar as dimensões das imagens, conforme feito no artigo que obteve a oitava colocação no desafio BACH de 2018 (KONÉ; BOULMANE, 2018). No primeiro momento foi feita a redução do tamanho para 399×299 pixels, com o objetivo de se manter a proporção original da imagem, evitando assim a distorção das características. Em seguida, foram cortados 50 pixels de cada uma das extremidades laterais da imagem, chegando ao resultado final de 299×299 pixels. Como consequência, houve uma grande redução no tamanho total do conjunto de dados que no primeiro momento possuía mais de 13

GB e, no segundo momento, o mesmo conjunto com 400 imagens passou a ocupar apenas 80 MB de espaço. Foi observado que outros trabalhos que participaram da competição também adotaram imagens de dimensões quadradas como entrada das abordagens, embora que em resoluções diferentes.

Em virtude do baixo número de amostras, foi necessária a implementação de técnicas de aumento de dados (do inglês, *data augmentation*) para se criar novas imagens com base nos exemplos existentes, porém modificando algumas características por meio de rotações, espelhamentos e alterações nos formatos. Com isso, a rede neural é apresentada a um número maior de imagens, aumentando a capacidade em generalizar as informações evitando o *overfitting*. Na Figura 14 podem-se observar exemplos das imagens que foram geradas para aumentar a quantidade de dados.

Figura 14 – Exemplo de imagens geradas pelo *ImageDataGenerator*.



Fonte: Produção do próprio autor.

Para implementar o aumento de dados foi escolhida a função *ImageDataGenerator* da biblioteca *keras* do *tensorflow* em Python, onde os parâmetros usados são apresentados abaixo e foram aplicados somente nos dados de treino.

- Rotação: 90°
- Zoom de até 5%
- Espelhamento vertical
- Espelhamento horizontal
- Deslocamento vertical: 0,2
- Deslocamento horizontal: 0,2

Além disso, os valores dos pixels das imagens foram padronizados pela média e desvio padrão do conjunto completo de imagens de forma que os dados tenham média zero e variância unitária. As pastas separadas, conforme cada classe, foram lidas pela biblioteca *flow_from_directory*. Para os dados de treino as imagens foram embaralhadas para diminuir a chance de *overfitting* durante o treinamento. A função separou em conjuntos (*batches*) com 20 imagens, que são carregadas na memória durante cada época de treinamento.

3.3 Rede CNN

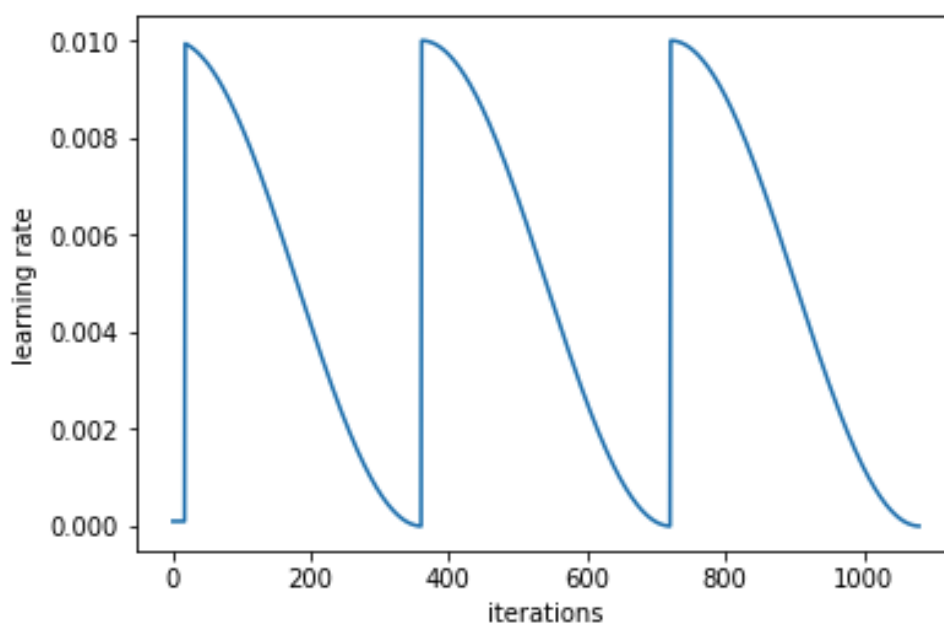
Tanto para a ResNet-50 quanto para a VGG19, foi realizado o *transfer learning* de seus modelos treinados com o conjunto de dados *ImageNet*, que possui 1000 classes distintas. Com isso, as configurações das redes foram modificadas para o conjunto de dados deste trabalho que possui apenas 4 classes. Assim, as camadas finais completamente conectadas (*dense*) foram substituídas por uma camada de achatamento (*flatten*), transformando os dados em um vetor de alta dimensão (geralmente da ordem de milhares de elementos). Na sequência, foi adicionada uma camada de *dropout* de 40%, camada esta importante para evitar o *overfitting*, pois tem como função remover de forma aleatória, durante o treinamento, conexões da camada densa com probabilidade de 40%, neste caso. Em seguida, foram adicionadas duas camadas densas com 128 e 64 neurônios e função de transferência *ReLU*, respectivamente, por fim, outra camada densa com 4 neurônios e função de ativação *softmax* para prever as classes dos dados.

Em virtude de o conjunto de dados possuir múltiplas classes, foi escolhida a função entropia cruzada categórica (do inglês, *Categorical Crossentropy*) para minimizar as perdas durante o treinamento. Já o otimizador utilizado foi o *Stochastic Gradient Descent* – *SGD* com uma taxa de aprendizado variável para evitar que o gradiente fique preso em um mínimo local e perca a capacidade de aprendizado.

Além disso, foi aplicada a técnica de *Early Stop* que consiste em definir um valor de paciência para que a rede demonstre melhora. Caso não haja avanço após certo número de épocas (o valor de paciência), o modelo interrompe o treinamento e utiliza os parâmetros que obtiveram os melhores resultados. A paciência escolhida foi de 100 épocas a rede VGG-19 e 40 épocas para a ResNet-50. A função escolhida para alterar a taxa de aprendizado ao longo

das épocas foi a função *Cosine Decay Restart*, cujo valor da taxa de aprendizado é reduzido ao longo das épocas e em dado momento é aumentado novamente (LOSHCHILOV, 2017). A Figura 15 demonstra como funciona a variação da taxa de aprendizado da função *Cosine Decay Restart*.

Figura 15 – Variação da taxa de aprendizado ao longo das épocas da função *Cosine Decay Restart*



Fonte: Wang, 2018.

Os valores dos hiperparâmetros do *Cosine Decay Restart* são responsáveis por definir a taxa de aprendizado que será reduzida até o valor mínimo, sendo calculado como a fração da taxa de aprendizado inicial multiplicada pelo parâmetro *alpha*. A taxa irá variar do máximo, o valor inicial, até o menor valor ao longo do número de degraus a partir do primeiro decaimento.

Os parâmetros para a VGG-19 foram:

- Taxa de aprendizado inicial: 0,001
- Degraus a partir do primeiro decaimento: 500
- *alpha*: 0,5
- Número máximo de épocas: 500
- Paciência: 100
- Monitor: *val_loss*

E para a *ResNet-50*:

- Taxa de aprendizado inicial: 0,01
- Degraus a partir do primeiro decaimento: 500
- *alpha*: 0,001
- Número máximo de épocas: 200
- Paciência: 40
- Monitor: *val_loss*

A definição dos parâmetros foi obtida após uma série de tentativas, com o objetivo de escolher os melhores valores de paciência e taxa de aprendizado, a fim de evitar que o erro da rede convergisse para valores pequenos de forma muito rápida. Fato responsável por indicar que o modelo está perdendo a capacidade de aprendizado e de generalização.

3.4 Extração de características

A rede convolucional que apresentar o melhor desempenho será utilizada para a extração de características para treinar os outros classificadores, na expectativa de potencializar a qualidade das informações que servirão de entrada para os demais classificadores. Para isso, a última camada totalmente conectada foi removida e substituída por uma camada de achatamento (*flatten*), transformando os dados em um vetor de alta dimensão (geralmente da ordem de milhares de elementos), que é obtido como saída da rede para cada imagem de entrada.

Complementarmente, será utilizada a técnica de Análise de Componentes Principais (do inglês, *Principal Component Analysis – PCA*) que tem como objetivo reduzir o número de variáveis de saída da rede, utilizando somente um conjunto reduzido de variáveis com o intuito de reduzir o custo computacional e memória utilizada, além de descartar dados que podem ser de baixa relevância. Para a redução de dimensionalidade, serão selecionadas as componentes responsáveis por explicar 95% da variância dos dados.

3.5 Modelo SVM

Partindo das características mais relevantes obtidas conforme a Seção 3.4, o modelo de Máquina de Vetores de Suporte (SVM) foi utilizado por meio da biblioteca *sklearn* do Python. Para a otimização dos hiperparâmetros, será utilizada a biblioteca *GridSearchCV* que testará diferentes configurações de parâmetros para a mesma base de dados, com o objetivo de encontrar a melhor combinação de valores e maximizar assertividade. Os parâmetros utilizados para o otimizador foram:

- *C*: 0,1; 0,5; 1; 2
- *gamma*: 0,1; 0,01; 0,0001; 0,00001
- *kernel*: rbf, poly, sigmoid, linear

3.6 Modelo XGBoost

Utilizando os mesmos dados de treino e técnicas da seção anterior, porém, o otimizador de hiperparâmetros utilizado na Seção 3.6 foi substituído por não obter bom desempenho com este modelo. Por isso, dessa vez foi utilizado a otimização Bayesiana, que otimiza os parâmetros conforme o desempenho de modelos anteriores e a probabilidade de evolução com base na combinação dos hiperparâmetros (SNOEK, J.; LAROCHELLE, H.; ADAMS, R, 2012). Após uma série de testes foram definidos o intervalo de valores para a otimização do classificador que deverá variar entre os seguintes intervalos:

- *max_depth*: (2; 10)
- *gamma*: (0,001; 5)
- *min_child_weight*: (5; 20)
- *subsample*: (0,4; 1.0)
- *colsample_bytree*: (0,4; 1.0)
- *learning_rate*: (0,001; 1.0)
- *num_boost_round*: (10; 100)

3.7 Recursos Computacionais

A base de dados utilizada é gratuita e pode ser obtida pelo site da conferência ICIAR 2018 sendo necessário se cadastrar previamente. Como principal biblioteca para as redes neurais

utilizou-se o *framework tensorflow* na versão 2.3.1 com o *backend* do Keras em sua versão 2.9.1. Além da rede neural, o Keras também foi utilizado para: *ImageDataGenerator* para o aumento de dados e *CosineDecayRestart* para o decaimento da taxa de aprendizado. O modelo XGBoost utilizado é o da biblioteca disponibilizada pelos autores no *github* (CHEN; GUESTRIN, 2022). Também foi utilizada a biblioteca *sklearn* para: utilização do modelo SVM, na busca automatizada de parâmetros por meio do algoritmo *GridSearchCV* e para a aplicação do PCA. Todo o código foi desenvolvido utilizando a linguagem Python na versão 3.8.2.

O hardware da máquina virtual fornecida pelo Google Colab foi: Processador Intel Xenon 2,30 GHz, 12 GB de RAM, espaço de armazenamento de 25 GB e placa gráfica Nvidia Tesla K80 com 12 GB de RAM.

4 RESULTADOS

Este capítulo tem como objetivo expor os resultados dos modelos propostos para que seja possível a comparação de desempenho entre eles. Para comparar o desempenho foram utilizadas as métricas de acurácia, precisão, sensibilidade e F_1 -score, além da matriz de confusão. Os códigos utilizados no desenvolvimento deste trabalho se encontram no *github* do autor (<https://github.com/antoniodragoc/Projeto-de-Graduacao-Antonio-Drago-Caetano>).

4.1 Resultados da VGG-19

Utilizando *transfer learning* de uma rede VGG-19 previamente treinada com a base de dados ImageNet e adaptando e treinando ela conforme descrito na Seção 3.3. Foram necessárias 234 épocas para o treinamento. A Figura 16 mostra a matriz de confusão com os resultados obtidos, enquanto a Tabela 3 destaca os valores de precisão, sensibilidade (*recall*) e F_1 -score separados por classe. A Tabela 4 mostra a acurácia e médias macros do modelo VGG-19.

Tabela 3 – Métricas por classe para a rede VGG-19

Classe	Precisão (%)	Sensibilidade (%)	F_1 -score (%)
Benigno	77,78	70,00	73,68
Não invasivo	70,00	70,00	70,00
Invasivo	80,00	80,00	80,00
Normal	90,91	100	95,24

Fonte: Produção do próprio autor.

Tabela 4 – Métricas Macro para a rede VGG-19

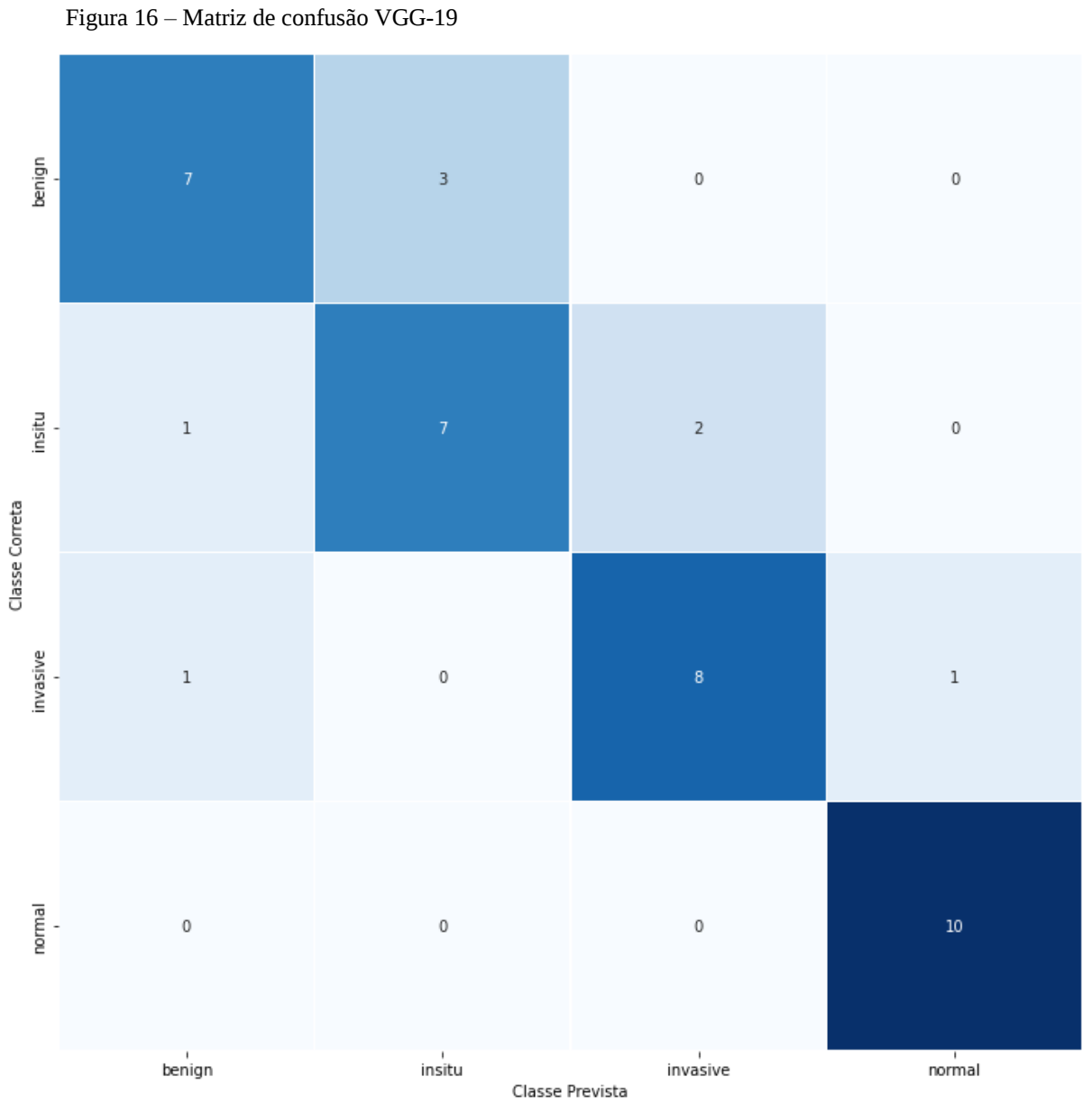
Acurácia (%)	Precisão (Macro) (%)	Sensibilidade (Macro) (%)	F_1 -score (Macro) (%)
80,00	79,67	80,00	79,84

Fonte: Produção do próprio autor.

Conforme observado na Tabela 4, todas as métricas se mantiveram próximas de 80% com destaque para as classes de tumor benigno e não invasivos que tiveram os valores de

sensibilidade impactados em razão de 30% dos dados destas classes terem sido classificados de forma incorreta, conforme pode ser visualizado na Figura 16.

Observa-se na Figura 16 que os maiores erros foram ocasionados devido ao modelo confundir tumor benigno com não invasivo e tumor não invasivo com tumor invasivo.



Fonte: Produção do próprio autor.

4.2 Resultados da ResNet-50

Em seguida foi utilizado *transfer learning* de uma rede ResNet-50 previamente treinada com a base de dados ImageNet e adaptada e treinada conforme descrito na Seção 3.3. Foram necessárias 104 épocas para treinar o modelo. A Figura 17 mostra a matriz de confusão com os resultados obtidos, enquanto a Tabela 5 destaca os valores de precisão, sensibilidade (*recall*) e *f₁-score* separados por classe. A Tabela 6 mostra a acurácia e médias macros da ResNet-50.

Tabela 5 – Métricas por classe para a rede ResNet-50

Classe	Precisão (%)	Sensibilidade (%)	F ₁ -score (%)
Benigno	75,00	90,00	81,82
Não invasivo	75,00	90,00	81,82
Invasivo	100	70,00	82,35
Normal	100	90,00	94,74

Fonte: Produção do próprio autor.

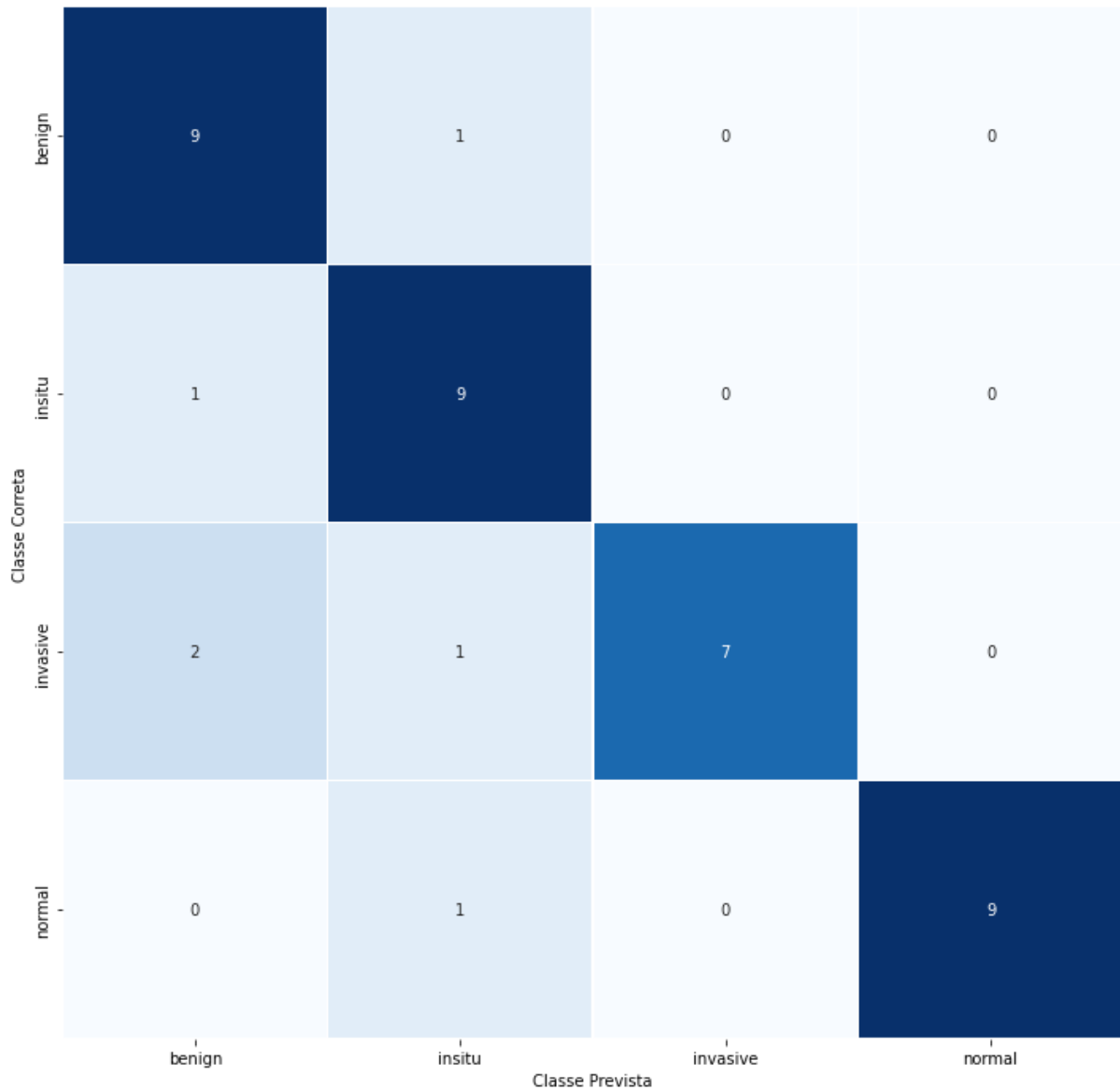
Tabela 6 – Métricas Macro para a rede ResNet-50

Acurácia (%)	Precisão (Macro) (%)	Sensibilidade (Macro) (%)	F ₁ -score (Macro) (%)
85,00	87,50	85,00	86,23

Fonte: Produção do próprio autor.

Conforme observado na Tabela 6, as métricas macros obtiveram bons resultados o que mostra que esta rede foi capaz de aprender com os dados apresentados no conjunto de treino. A classe que teve o pior desempenho foi a de câncer invasivo que foi classificada de forma errada em 3 ocasiões, 2 delas como benigno e uma como não invasiva, conforme visto na Figura 17.

Figura 17 – Matriz de confusão ResNet-50



Fonte: Produção do próprio autor.

Devido ao fato da ResNet-50 ter apresentado melhor resultado que a VGG-19, ela foi usada como extratora de características para os demais classificadores. O PCA foi aplicado sobre a saída dela, conforme Seção 3.4, reduzindo um vetor de 204800 elementos para um vetor de características de 269 elementos por imagem.

4.3 Resultados da ResNet-50 + SVM

A SVM foi treinada com as informações geradas pela rede ResNet-50 que leu o conjunto de dados de treino e gerou um vetor com características das imagens. Para reduzir o custo computacional do classificador, foi aplicada a técnica de análise de componentes principais na saída da rede neural, seguida por uma etapa de busca dos melhores hiperparâmetros pela técnica *GridSearchCV* que avalia diversos valores de hiperparâmetros a fim de maximizar o desempenho. O modelo com melhor desempenho obtido foi composto pelos hiperparâmetros: $C = 1$, $\gamma = 0,00001$ e o *kernel* sigmoid. A Figura 18 traz a matriz de confusão deste modelo e a Tabela 7 as métricas de desempenho de cada classe. A Tabela 8 mostra a acurácia e médias macros do modelo das máquinas de vetores de suporte.

Tabela 7 – Métricas por classe para o modelo ResNet-50 + SVM

Classe	Precisão (%)	Sensibilidade (%)	F ₁ -score (%)
Benigno	81,82	90,00	85,71
Não invasivo	75,00	90,00	81,82
Invasivo	100	70,00	85,35
Normal	100	100	100

Fonte: Produção do próprio autor.

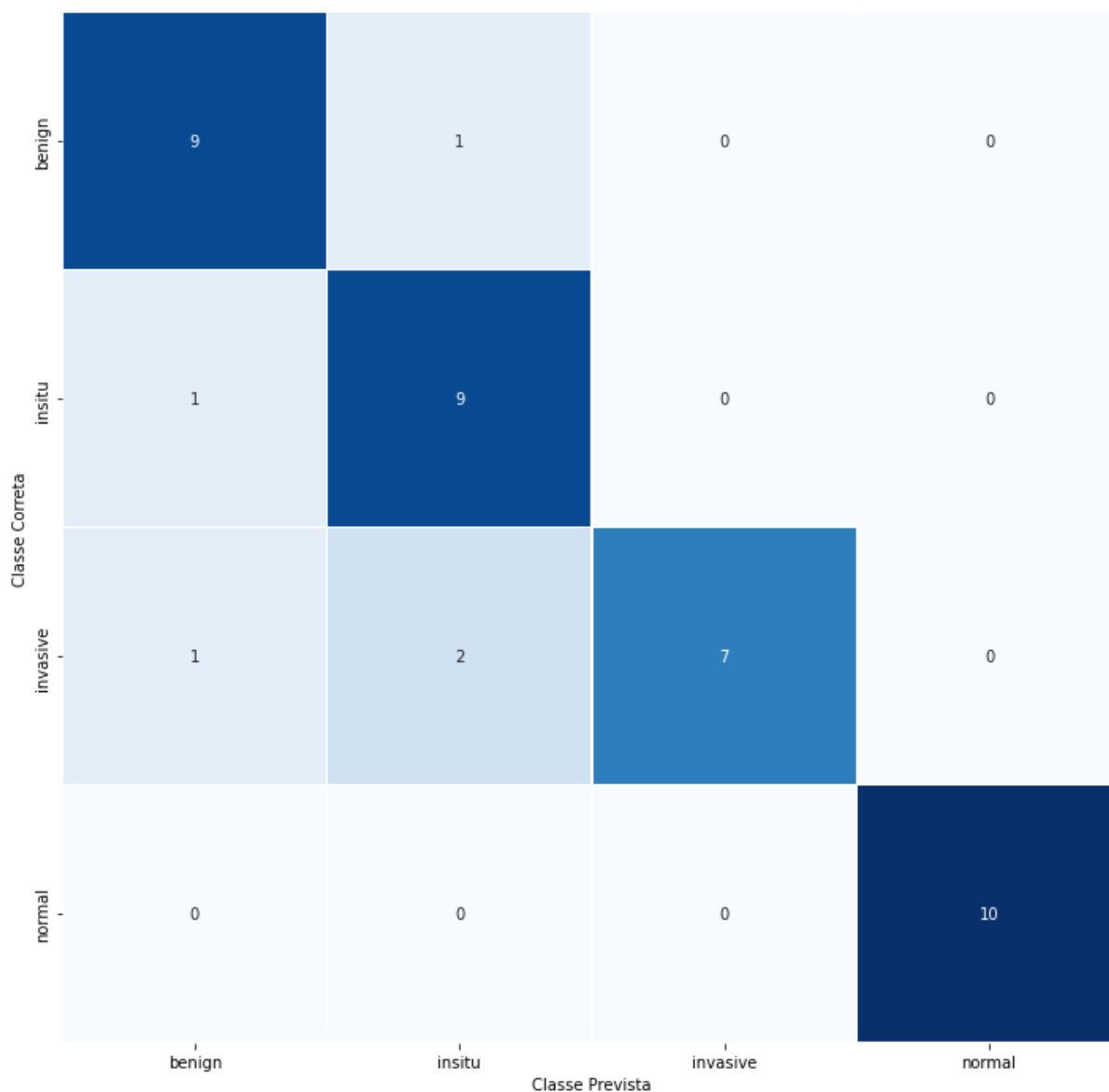
Tabela 8 – Métricas Macro para a rede ResNet-50 + SVM

Acurácia (%)	Precisão (Macro) (%)	Sensibilidade (Macro) (%)	F ₁ -score (Macro) (%)
87,50	89,20	87,50	88,34

Fonte: Produção do próprio autor.

Pela Tabela 8 pode-se observar que este classificador obteve bons resultados na maioria das classes, errando apenas em 12,5% das ocasiões. Na Figura 18, percebe-se que a classe que teve o desempenho mais prejudicado foi a invasiva, que apresentou 3 classificações incorretas (2 sendo classificadas como não invasiva e 1 como benigna), reduzindo o valor das métricas macros.

Figura 18 – Matriz de confusão ResNet-50 + SVM



Fonte: Produção do próprio autor.

4.4 Resultados da ResNet-50 + XGBoost

Assim como no SVM foram utilizadas as informações geradas pela rede ResNet-50, que leu o conjunto de dados de treino e gerou um vetor com características das imagens. Para reduzir o custo computacional do classificador, também se aplicou a técnica de análise de componentes principais, porém a otimização foi feita de forma diferente. Neste caso se utilizou o método de otimização Bayesiana que consiste na otimização conforme a probabilidade de se obter o melhor desempenho para o modelo. O melhor conjunto obtido foi composto pelos valores

$max_depth = 7$, $gamma = 0,05$, $min_child_weight = 13,2501$, $subsample = 1$, $colsample_bytree = 1$, $learning_rate = 0,6422$ e $num_boost_round = 55,6894$. A Figura 19 apresenta a matriz de confusão deste modelo e na Tabela 9 estão as métricas de cada classe. A Tabela 10 mostra a acurácia e médias macros do modelo ResNet-50 + XGBoost.

Tabela 9 – Métricas por classe para o modelo ResNet-50 + XGBoost

Classe	Precisão (%)	Sensibilidade (%)	F ₁ -score (%)
Benigno	87,50	70,00	77,78
Não invasivo	63,64	70,00	66,67
Invasivo	53,85	70,00	60,87
Normal	100	80,00	88,89

Fonte: Produção do próprio autor.

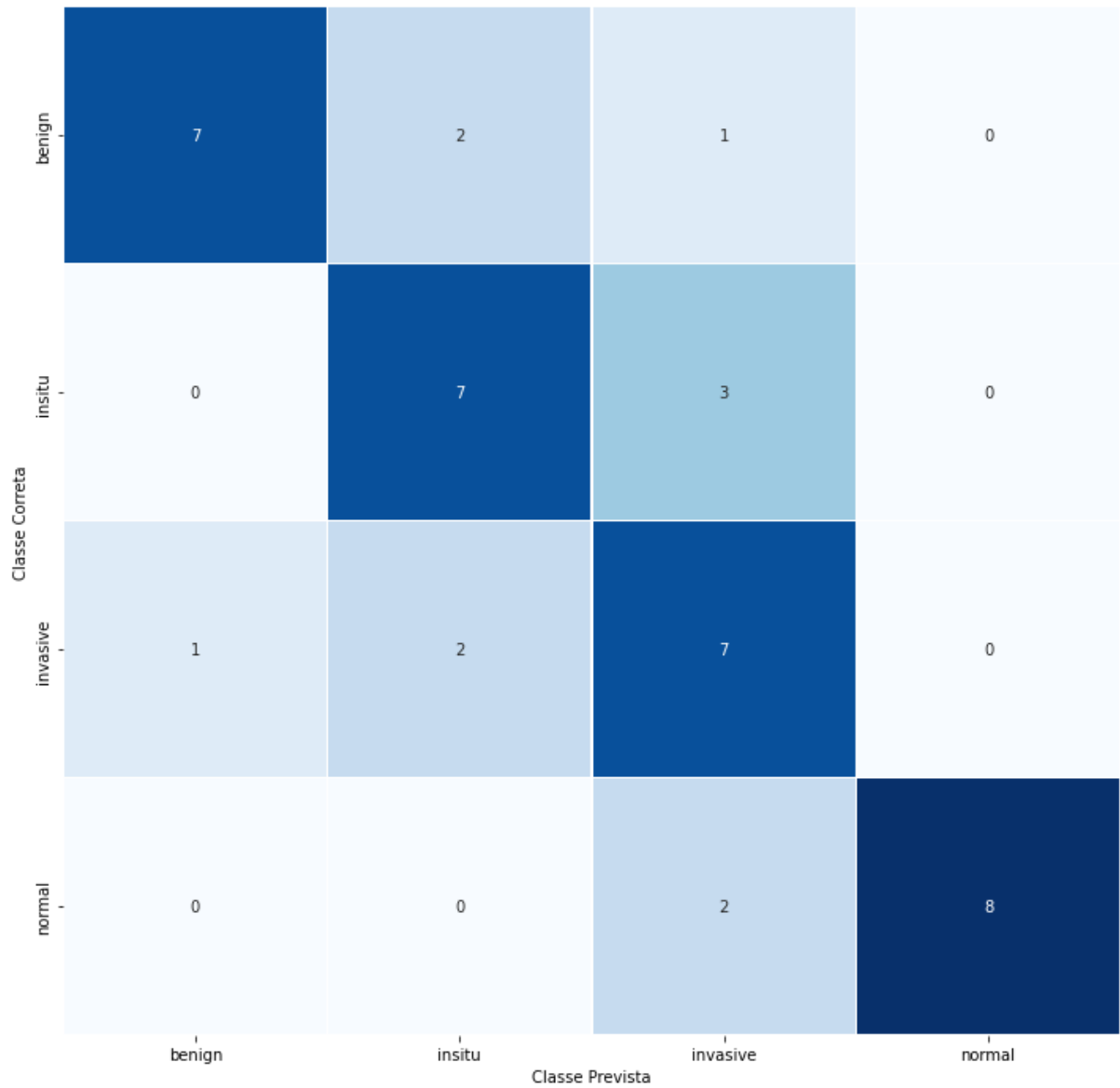
Tabela 10 – Métricas Macro para a rede ResNet-50 + XGBoost

Acurácia (%)	Precisão (Macro) (%)	Sensibilidade (Macro) (%)	F ₁ -score (Macro) (%)
72,50	76,25	72,50	74,33

Fonte: Produção do próprio autor.

Conforme observado na Tabela 10, os resultados macros foram inferiores aos modelos anteriores, porém ainda assim superiores a 70%, o que mostra bom desempenho considerando o conjunto reduzido de dados. Neste caso, a distribuição de acertos se deu de forma equilibrada. O maior destaque se deu nos casos classificados pela técnica como câncer invasivo, na qual foram classificadas 6 imagens erroneamente, seguida pelos erros de classificação como classe não invasiva, com 4 imagens classificadas de forma errada. Conforme visto na Figura 19.

Figura 19 – Matriz de confusão ResNet-50 + XGBoost



Fonte: Produção do próprio autor.

4.5 Comparação dos resultados

Nesta etapa serão agrupadas as acurácias dos modelos ordenados do maior para o menor, conforme mostrado na Tabela 11.

Tabela 11 – Comparação dos modelos de classificação.

Modelo	Acurácia (%)	Precisão (Macro) (%)	Especificidade (Macro) (%)	F1-score (Macro) (%)
ResNet-50 + SVM	87,50	89,20	87,50	88,34
ResNet-50	85,00	87,50	85,00	86,23
VGG-19	80,00	79,67	80,00	79,84
ResNet-50 + XGBoost	72,50	76,25	72,50	74,33

Fonte: Produção do próprio autor.

Quando as redes convolucionais utilizadas, ResNet-50 e VGG-19, são comparadas percebe-se que a primeira obteve melhores resultados em todos os aspectos analisados. Além disso, observando a primeira linha da Tabela 11 ressalta a importância de utilizar mais de uma metodologia para abordar o problema, pois somando a capacidade da rede ResNet-50 de extrair características das imagens e a habilidade das máquinas de vetores de suporte de separar características resultou em um modelo com melhor desempenho. Por outro lado, o mesmo não foi observado nos resultados da XGBoost.

De forma a complementar a análise realizada, a Tabela 12 apresenta os 5 melhores modelos do desafio BACH de 2018 (ARESTA; ARAÚJO; KWOK; et al., 2019) e o melhor resultado obtido neste trabalho. É importante ressaltar que, embora os dados utilizados para o treinamento das técnicas tenham sido os mesmos (se for desconsideradas as divisões dos dados), os dados utilizados para teste neste trabalho são diferentes dos dados de testes utilizados na competição. Isto ocorreu porque estes não foram publicamente disponibilizados, e a equipe participante da competição deveria enviar o modelo treinado para avaliação.

Pode-se observar na Tabela 12 que o resultado obtido neste trabalho (ResNet-50 + SVM) é compatível com os melhores resultados obtidos na literatura. Embora esta não seja uma

comparação em condições de igualdade, ela indica que a abordagem utilizada neste trabalho tem um potencial semelhante aos das melhores abordagens propostas na competição.

Tabela 12 – Comparação do melhor modelo obtido com top-5 resultados do desafio

Modelo	Acurácia (%)	Tamanho da Imagem
ResNet-50 + SVM	87,50	299x299
Resnet-101; Densenet-161	87,00	224x224
Inception-Resnet-v2	87,00	299x299
Resnet-34, 50, 101	86,00	308x308 e 615x615
Inception-v3	84,00	512x512
Densenet-161	83,00	205x154

Fonte: Aresta et al (2019).

Nota: Adaptado pelo autor.

5 CONCLUSÕES E TRABALHOS FUTUROS

5.1 Conclusões

O presente trabalho teve como objetivo apresentar um estudo com ferramentas de aprendizado de máquina mesclando diferentes metodologias e ferramentas, além da reutilização de técnicas como forma de entrada de dados para outros modelos. Tal objetivo foi bem sucedido por meio da utilização da aplicação de diferentes técnicas de classificação combinadas ao uso de *data augmentation* para o treinamento dos modelos.

As redes convolucionais obtiveram bons desempenhos, 80% e 85% para as redes VGG-19 e ResNet-50, respectivamente. Além disso, a utilização das redes neurais como extratores de características para o treinamento de classificadores clássicos se mostrou bem eficaz, resultando nas acurácias de 87,5% e 72,5% para os modelos SVM e XGBoost, respectivamente.

A rede ResNet-50 foi escolhida para a etapa de extração de características por conta da sua maior capacidade de generalização com o conjunto de dados reduzido. Para isso, algumas camadas foram ajustadas para o desafio proposto além de seus hiperparâmetros otimizados para o problema em questão. A etapa que tomou mais tempo de execução e de pesquisa foi de *fine tuning* dos modelos de redes convolucionais, já que o grande número de camadas das redes aumenta o tempo de treinamento.

Por fim, é também relevante se observar a matriz de confusão dos dados de teste para analisar o desempenho do modelo em cada classe e com isso observar a capacidade de generalização do modelo.

5.2 Trabalhos futuros

Com o intuito de aprimorar os resultados pode-se sugerir a implementação de outros tipos de redes neurais como a ResNet-152, Inception e Google Net, assim como avaliar outras abordagens de treinamento das redes convolucionais, como funções de perda, algoritmos de treinamento e de ajuste da taxa de aprendizado, técnicas de *fine tuning*, *early stop* e *data*

augmentation para buscar obter resultados melhores. Também avaliar outras técnicas para reduzir a dimensão no lugar do PCA, já que este é um método não supervisionado.

Outra sugestão de melhoria é buscar formas de diversificar o conjunto de dados mesclando com fontes que tenham imagens semelhantes para complementar na hora de treinar os modelos e, com isso, aumentar a variedade nas informações apresentadas aos modelos.

REFERÊNCIAS BIBLIOGRÁFICAS

- ALKHALEEF, M.; WU, C. A Hybrid CNN and RBF-Based SVM Approach for Breast Cancer Classification in Mammograms, **IEEE International Conference on Systems, Man, and Cybernetics (SMC)**, pp. 894-899, 2018.
- ARAÚJO, T.; ARESTA, G.; CASTRO, E.; ROUCO, J.; et al, Classification of breast cancer histology images using Convolutional Neural Networks. **PLoS ONE**, 12(6): e0177544., 2017
- ARESTA, G.; ARAÚJO, T.; KWOK, S., BACH: grand challenge on breast cancer histology images. **ELSEVIER**, 2019
- BENHAMMOU, Y.; TABIK, S.; ACHCHAB, B.; HERRERA, F. A first study exploring the performance of the state-of-the art CNN model in the problem of breast cancer, **Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications**, pp. 1-6, 2018.
- BISHOP, C.; NASRABADI, N. Pattern recognition and machine learning, vol 4. No.4. **Springer**, 2006.
- Breast cancer. **World Health Organization**, 2021. Disponível em: <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>. Acesso em 06 de março de 2022.
- BREIMAN, L.; FRIEDMAN, J.; OLSHEN, A.; STONE, C. *Classification And Regression Trees*, 1st ed. **Routledge**, 1984.
- BROWNLEE. J. Better Deep Learning. **Machine Learning Mastery**, 2019.
- CAMPBELL, C; YING, Y: Learning with support vector machines. **Synthesis lectures on artificial intelligence and machine learning**, 2011.
- CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. **ArVix**. San Francisco, 2016. Disponível em: <https://arxiv.org/abs/1603.02754>. Acesso em 25 de março de 2022.
- CHEN, T.; GUESTRIN, C. XGBoost: eXtreme Gradient Boosting. Disponível em: <https://github.com/dmlc/xgboost>. Acesso em 25 de março de 2022.
- CORTES, C; VAPNIK, V, Support-Vector Networks. **Machine Language**, v. 20, n. 3, p. 273-297, set. 1995.
- F.-F. ImageNet: A large-scale hierarchical image database. **IEEE Conference on Computer Vision and Pattern Recognition**, pp. 248-255, 2009.
- FRIEDMAN, J. *Greedy Function Aproximation: A Gradient Boosting Machine*. **IMS**. Fev. 1999.

Grand Challenge on BreAst Cancer Histology images – BACH. **ICIAR 2018**, 2018. Disponível em: <https://iciar2018-challenge.grand-challenge.org/>. Acessado em 01 de Agosto de 2022.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning. **MIT press**, 2019

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep Residual Learning for Image Recognition. **ArVix**. Disponível em: <https://arxiv.org/abs/1512.03385v1>. Acesso em: 05 de março de 2022.

HEYAM, H.; AL-BAITY; ALGHUNAIM, S., On the scalability of machine-learning algorithms for breast cancer prediction in big data contexto, **IEEE Access**, vol.7, pp.91535 – 91546, 2019.

KARPATHY, A. et al. Cs231n convolutional neural networks for visual recognition. **Stanford**. Disponível em: <http://cs231n.stanford.edu/2016/>. Acesso em 01 de junho de 2022.

KIM, D.; ARAUJO, A.; TSAI, A; et al. Saber é prevenir: uma nova abordagem no combate ao câncer de mama, **Ciência & Saúde Coletiva**, 2021. 15(Supl. 1):1377-1381, 2010.

KONÉ, I.; BOULMANE, L. Hierarchical ResNeXt Models for Breast Cancer Histology Image Classification. **ArVix**. Springer, 2018. Disponível em: <https://arxiv.org/abs/1810.09025>. Acesso em 06 de março de 2022.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, v. 521, 2018.

LOSHCHILOV, I.; HUTTER, F. SGDR: Stochastic Gradient Descent with Warm Restarts. **ArVix**. Freiburg, 2017. Disponível em: <https://arxiv.org/abs/1608.03983>. Acesso em 06 de março de 2022.

MARKOULIDAKIS, I.; RALLIS, I.; GEORGOULAS, I.; KOPSIAFTIS, G.; DOULAMIS, A.; DOULAMIS, N. Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem. **Technologies** 2021, 9, 81. 2021

MURPHY, K. *Machine Learning: A Probabilist Perspective*. **MIT Press**, 2012.

PAN, J.; KWOK, T.; YANG, Q. Transfer learning via dimensionality reduction. **AAAI Press**, p.677 – 682. Chicago: 2008.

RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATHY, A.; KHOSLA, A.; BERNSTEIN, M. S.; BERG, A. C.; LI, SAEED, M. An Introduction to Recurrent Neural Networks And The Math That Powers Them. **machinelearningmastery**, 2021. Disponível em: <https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/>. Acesso em 06 de março de 2022.

SIMONYAN, K.; ZISSERMAN. A. Very Deep Convolutional Networks for Large-Scale Image Recognition. **ArVix**. Disponível em: <https://arxiv.org/abs/1409.1556>. Acesso em: 05 de março de 2022.

SNOEK, J.; LAROCHELLE, H.; ADAMS, R. PRACTICAL BAYESIAN OPTIMIZATION OF MACHINE LEARNINGALGORITHMS. **ArVix**. Toronto, 2012. Disponível em: <https://arxiv.org/abs/1206.2944>. Acesso em 06 de março de 2022.

STRAPASSON, E.; VENCovsky, R.; BATISTA, R. Seleção de Descritores na Caracterização de Germoplasma de Paspalum sp. por meio de Componentes Principais. **Rev. bras. zootec.**, 29(2):373-381, 2000.

THANI, I.; KASBE, T. Breast Cancer: State –of- the-art causes and diagnosis, 2nd International Conference on Data, **Engineering and Applications (IDEA)**, pp. 1-6, 2020.

THEODORIDIS, S; KOUTROUMBAS, K: *Pattern Recognition*, 4TH ed. **Academic Press**, 2009.

WANG, C. ,2018. Disponível em: <https://towardsdatascience.com/https-medium-com-reina-wang-tw-stochastic-gradient-descent-with-restarts-5f511975163>. Acesso em 22 de julho de 2022.

ZEILER, M.; FERGUS, R.: Visualizing and understanding convolutional networks. **Computer Vision ECCV 2014**. pp. 818–833, 2014