

**UNIVERSIDADE DE SÃO PAULO**  
**FACULDADE DE CIÊNCIAS FARMACÊUTICAS**  
**Curso de Graduação em Farmácia-Bioquímica**

**USO DE INTELIGÊNCIA ARTIFICIAL PARA AUXÍLIO DIAGNÓSTICO**  
**DIFERENCIAL DE VARÍOLA DOS MACACOS**

**João Gabriel Nascimento Castilho**

Trabalho de Conclusão do Curso de  
Farmácia-Bioquímica da Faculdade de  
Ciências Farmacêuticas da  
Universidade de São Paulo

Orientador(a):  
Dr. Mauro César Cafundó de Moraes - FCF / USP - CSBL / USP

São Paulo

2023

## SUMÁRIO

<b>SUMÁRIO.....</b>	<b>1</b>
<b>LISTA DE ABREVIATURAS E SIGLAS.....</b>	<b>2</b>
<b>LISTA DE FIGURAS.....</b>	<b>3</b>
<b>LISTA DE TABELAS.....</b>	<b>3</b>
<b>RESUMO.....</b>	<b>5</b>
<b>1.INTRODUÇÃO.....</b>	<b>6</b>
<b>2.OBJETIVOS.....</b>	<b>8</b>
<b>3. MATERIAL E MÉTODOS.....</b>	<b>9</b>
<b>3.1. ANÁLISE ESTATÍSTICA.....</b>	<b>12</b>
<b>3.1.1 ACURÁCIA.....</b>	<b>13</b>
<b>3.1.2 PRECISÃO.....</b>	<b>14</b>
<b>3.1.3 RECALL (SENSIBILIDADE).....</b>	<b>14</b>
<b>3.1.4 F1-SCORES.....</b>	<b>15</b>
<b>4. RESULTADOS.....</b>	<b>15</b>
<b>5. DISCUSSÃO.....</b>	<b>23</b>
<b>6. CONCLUSÕES.....</b>	<b>24</b>
<b>7. REFERÊNCIAS.....</b>	<b>25</b>
<b>8. APÊNDICES.....</b>	<b></b>

## LISTA DE ABREVIATURAS E SIGLAS

<b>CSBL</b>	Computational Systems Biology Laboratory
<b>USP</b>	Universidade de São Paulo
<b>OMS</b>	Organização Mundial de Saúde
<b>Mpox</b>	Vírus conhecido anteriormente como <i>Monkeypox</i>
<b>CNN</b>	Rede Neural Convolucional
<b>CDC</b>	Centros de Controle e Prevenção de Doenças
<b>PCR</b>	Reação em Cadeia da Polimerase
<b>qPCR</b>	Reação em Cadeia da Polimerase em Tempo Real
<b>HIAE</b>	Hospital Israelita Albert Einstein
<b>HC FMUSP</b>	Hospital das Clínicas da Faculdade de Medicina da USP
<b>API</b>	Interface de Programação de Aplicação
<b>Curva COR</b>	Curva Característica de Operação do Receptor
<b>VP</b>	Verdadeiro Positivo
<b>VN</b>	Verdadeiro Negativo
<b>FP</b>	Falso Positivo
<b>FN</b>	Falso Negativo
<b>OvR</b>	<i>One versus Rest</i>
<b>R</b>	Recall
<b>P</b>	Precisão
<b>ASC</b>	Área Sob a Curva

## LISTA DE FIGURAS

Fig.1. A) Esquema de Pré-Processamento de Imagens

Fig.1. B) Esquema de Treinamento do Modelo com uso de Imagens Rotuladas

Fig.1. C) Esquema de Reconhecimento de Lesão através do Modelo previamente treinado

Fig.2 Esquema da Validação Cruzada 5-Fold, indicando as divisões do *Dataset*

## LISTA DE TABELAS

Tabela 1 - Quantidade de Fotos e Imagens Rotuladas no *Dataset*

Tabela 2 - Matriz de Confusão Binária

Tabela 3 - Relação entre resultado predito do treinamento através do modelo “VGG-16” e resultado esperado

Tabela 4 - Relação entre resultado predito do treinamento através do modelo “VGG-19” e resultado esperado

Tabela 5 - Relação entre resultado predito do treinamento através do modelo “Xception” e resultado esperado

Tabela 6 - Relação entre resultado predito do treinamento através do modelo “ResNet50” e resultado esperado

Tabela 7 - Relação entre resultado predito do treinamento através do modelo “MobileNet\_V2” e resultado esperado

Tabela 8 - Relação entre resultado predito do treinamento através do modelo “Inception\_V3” e resultado esperado

Tabela 9 - Relação entre resultado predito do treinamento através do modelo “Dense\_201” e resultado esperado

Tabela 10 - Relação entre resultado predito pela validação através do modelo “VGG-16” e resultado esperado

Tabela 11 - Relação entre resultado predito pela validação através do modelo “VGG-19” e resultado esperado

Tabela 12 - Relação entre resultado predito pela validação através do modelo “Xception” e resultado esperado

Tabela 13 - Relação entre resultado predito pela validação através do modelo “ResNet50” e resultado esperado

Tabela 14 - Relação entre resultado predito pela validação através do modelo “MobileNet\_V2” e resultado esperado

Tabela 15 - Relação entre resultado predito pela validação através do modelo “Inception\_V3” e resultado esperado

Tabela 16 - Relação entre resultado predito pela validação através do modelo “Dense\_201” e resultado esperado

Tabela 17 - Métricas de desempenho da validação dos diferentes modelos, análise binária

Tabela 18 - Métricas de desempenho da validação dos diferentes modelos, análise multiclasse one vs rest

## RESUMO

**Palavras-chave:** *Varíola dos macacos, aprendizado de máquina, rede neural, diagnóstico com auxílio de computador.*

**Introdução:** Em Julho de 2022 a Organização Mundial da Saúde (OMS) anunciou a Varíola dos Macacos, causada pelo vírus mpox (conhecido anteriormente como *Monkeypox*), como emergência de saúde pública internacional. Uma das dificuldades do controle da doença é sua sintomatologia ser similar a doenças, como Varíola e Varicela, o que dificulta um diagnóstico rápido e eficaz. Nos últimos anos o desenvolvimento e a aplicação de técnicas de aprendizado de máquina (*Machine Learning*) ganharam destaque, pois apresentaram formas de aumentar a confiabilidade, previsibilidade e performance no diagnóstico de doenças, através da análise de informações clínicas. O uso desta tecnologia para a análise de imagens de lesões de pele da Varíola dos Macacos e de outras doenças que geram erupções cutâneas similares, pode permitir que padrões não reconhecíveis por dermatologistas sejam identificados, auxiliando o diagnóstico diferencial de maneira eficiente.

**Objetivos:** Desenvolver método para reconhecimento de lesões causadas por Varíola dos Macacos através de aprendizado de máquina. Diferenciar lesões causadas por Varíola dos Macacos de outras doenças que apresentam sintomas similares apenas por análise de imagens de pele do paciente.

**Material & Métodos:** Uso de modelo computacional de Rede Neural Convolucional (CNN) para a análise de imagens previamente processadas de lesões de pele de Varíola dos Macacos, Varicela, Herpes Simples, Impetigo, Sífilis e Molusco Contagioso, obtidas de bancos de dados públicos. E posterior análise estatística dos resultados obtidos e desempenho do algoritmo.

**Resultados:** Foi relacionado os rótulos verdadeiros com os rótulos preditos no *Dataset* de treinamento e de validação para se obter os resultados, de forma binária e de forma multiclasse *One versus Rest*, para os diferentes modelos.

**Conclusão:** O método computacional desenvolvido não obteve resultados desejados, devido ao tamanho limitado do *Dataset*, mas apresentou potencial para estudos futuros sobre a utilização de métodos computacionais para realização de diagnósticos.

## 1. INTRODUÇÃO

A Varíola dos Macacos (Mpox), doença causada por um vírus do gênero *Orthopoxvirus*, da família *Poxviridae*, é conhecida desde a década de 1970, quando foi identificada pela primeira vez em um humano. Até o início de 2018, apresentou aproximadamente 600 casos confirmados. Por se tratar de uma doença negligenciada, muitos outros casos não foram confirmados, devido a subnotificação (SKLENOVSKÁ; VAN RANST, 2018)

Contudo, a atenção devida a esta enfermidade foi dada apenas em 2022, quando a Varíola dos Macacos começou a ser reportada em muitos locais ao redor do mundo. Dos 110 países que apresentaram casos no ano de 2022, 103 não possuíam histórico da doença. E também o número de casos reportados foi muito acima do comum, 84.318 casos, sendo o desfecho de 110 destes casos a morte do paciente, segundo o Centros de Controle e Prevenção de Doenças dos Estados Unidos (CDC) (2022 Mpox Outbreak Global Map, acessado em 5 de Janeiro de 2023). Isso fez com que em 23 de Julho de 2022, a Mpox fosse categorizada pela OMS (Organização Mundial da Saúde) como uma Emergência de Saúde Pública de Âmbito Internacional.

O vírus ativa a resposta do sistema imunológico, portanto, alguns dos sintomas dessa doença são febre, aumento dos linfonodos, e dores musculares. Porém, aproximadamente 3 dias depois do aparecimento da febre, começam a surgir erupções cutâneas, que evoluem com o passar dos dias, até formarem crostas e, ao caírem, podem deixar cicatrizes. (KALER et al., 2022)

A respeito da transmissão, ocorre de roedores, os hospedeiros principais, para seres humanos, sendo a transmissão de humano para humano, mais difícil (KALER et al., 2022), contudo este surto recente demonstrou que, por mais incomum que seja, esta forma de infecção pode causar um número de casos alarmante. Que além de significar um aumento no número de enfermos e mortos,

também pode apontar para a aparições de mutações, que, mesmo tendo pequenas chances de acontecerem, podem levar a variantes do vírus que sejam mais perigosas para a população (KOZLOV, 2022b).

Isto exposto, mostra-se necessário que exista um controle da doença, e devido a uma quantidade limitada de vacinas e medicamentos para o tratamento (KOZLOV, 2022a), este controle deve ser feito através da identificação de casos, para que possam ser devidamente isolados, assim evitando posterior transmissão. O método atual para se diagnosticar casos de Mpox é, a partir de uma suspeita de um profissional da saúde ao ver lesões de pele no paciente, realizar testes para diferenciar esta doença de outras que causam sintomas de pele, como Herpes Simples, Varicela, ou Sífilis. Contudo estes testes de especificidade, como PCR, qPCR ou análise por microscopia eletrônica, demandam tecnologias e recursos com preço elevado, o que é problemático levando em conta que a Varíola dos Macacos é endêmica em muitos países com dificuldade ao acesso a estes recursos (HRAIB et al., 2022).

Nos últimos anos, o uso de aprendizado de máquina (*Machine Learning*) dentro da área da saúde vem se expandindo, pois se mostrou uma forma inovadora de desenvolver novos métodos para a resolução de problemas. Uma das principais formas na qual esta tecnologia foi utilizada para auxiliar na realização de diagnóstico de doenças, de maneira mais eficiente e confiável, foi através da análise de imagens por computadores (SHEHAB et al., 2022). O fato da Mpox apresentar um sintoma facilmente registrável em forma de imagem, o uso da inteligência artificial pode apresentar uma nova forma de auxiliar no diagnóstico da doença de forma específica, sem a necessidade de testes de alto custo.

Propriedades que serão levadas em consideração para a criação deste trabalho serão a explicabilidade e a interpretabilidade do modelo computacional. Este primeiro ponto se refere a capacidade do código de apresentar com clareza de que forma obtiveram-se os resultados, demonstrando os dados e os algoritmos utilizados para a tomada de decisão. (DOSHI-VELEZ; KIM, 2017) Já a interpretabilidade trata da propriedade que o modelo tem de apresentar resultados que possam ser interpretados por especialistas da área. (BURKART; HUBER, 2021) Neste projeto em questão, essas duas propriedades são importantes para que médicos possam garantir que os resultados obtidos pela rede são confiáveis e



poderão ser usados para a tomada de decisões. Além disso, a explicabilidade e a interpretabilidade permitem que possíveis vieses do modelo sejam identificados. (VELLIDO, 2019)

#### JUSTIFICATIVA:

Apesar de historicamente apresentar um pequeno número de casos, e de ter deixado de ser considerado como emergência de saúde global pela OMS (Organização Mundial da Saúde) em 11 de Maio de 2023, o aumento repentino que ocorreu no meio do ano de 2022 demonstrou que a Varíola dos Macacos tem o potencial epidemiológico para causar uma próxima grande pandemia. Portanto, deve ser devidamente notificada e controlada, principalmente considerando uma escassez em tratamentos até o momento.

O desenvolvimento de métodos no auxílio diagnóstico, através da tecnologia do aprendizado de máquina, é imprescindível para este objetivo, pois pode significar uma maneira precisa e menos custosa para se diferenciar a Varíola dos Macacos de outras doenças. Como consequência, pode-se prevenir a ocorrência de novos surtos, independente do país, assim impedindo possíveis mutações e que vidas sejam perdidas.

Em paralelo, este projeto possibilita avanços na área da saúde, através da aplicação de novas tecnologias de inteligência artificial, e pode servir de base para novas pesquisas sobre o assunto no futuro.

## 2. OBJETIVOS(S)

Este trabalho tem como objetivo construir um modelo computacional, baseado em *Deep Learning*, método de aprendizado de máquina que utiliza múltiplas camadas para transformação dos dados, para identificar lesões de pele em imagens tiradas de pacientes com Varíola dos Macacos. E através do modelo desenvolvido, diferenciar feridas ocasionadas pela Mpox das lesões de pele causadas por Herpes Simples, Sífilis, Varicela, Molusco Contagioso ou Impetigo, doenças com uma sintomatologia similar.

### 3. MATERIAL E MÉTODOS

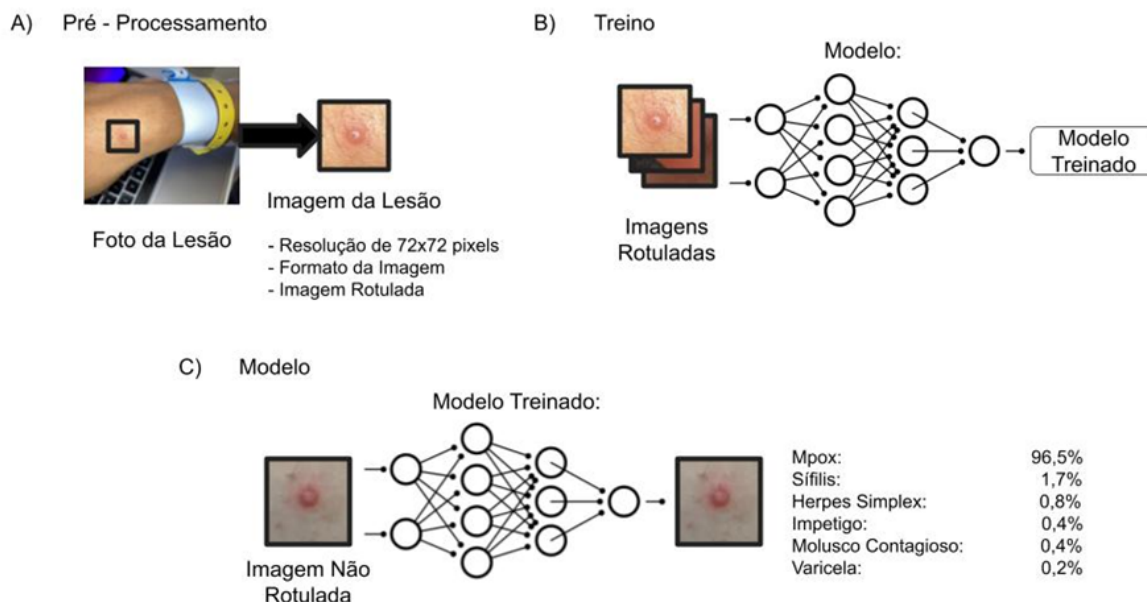
Para o desenvolvimento do algoritmo será utilizado um banco de imagens das lesões de pele já rotuladas com as respectivas doenças. Neste trabalho, as fotos de lesões da Varíola dos Macacos foram cedidas pelos próprios pacientes, mediante pedido médico, e enviadas por clínica especializada do Hospital Israelita Albert Einstein (HIAE) e Hospital das Clínicas da Faculdade de Medicina da Universidade de São Paulo (HC, FMUSP). As imagens das outras lesões foram retiradas de atlas dermatológicos online, o Hellenic Dermatological Atlas e o Dermatology Atlas (Acessado em 20 de julho de 2022).

**Tabela 1** - Quantidade de Fotos e Imagens Rotuladas no *Dataset*

Rótulo	Nº de Fotos	Nº de Imagens
Herpes simplex	88	99
Impetigo	95	107
Molluscum contagiosum	149	125
Sífilis	220	197
Varicela	63	116
Varíola dos Macacos	35	57
Total	650	701

**FONTE:** Elaborado pelo autor

**Fig.1. A)** Esquema de Pré-Processamento de Imagens **B)** Esquema de Treinamento do Modelo com uso de Imagens Rotuladas **C)** Esquema de Reconhecimento de Lesão através do Modelo previamente treinado



**FONTE:** Elaborado pelo autor

Para padronizar as fotos, que foram adquiridas de diferentes aparelhos celulares, fizemos um pré-processamento das imagens digitais, alterando o tipo de dado do arquivo, resolução e rotulando de acordo com a doença (Figura 1.A). Além disso, o pré-processamento também é necessário para aumentar a generalidade do modelo, minimizando possíveis vieses causados pela montagem do *Dataset* (Tabela 1), além de, por questões éticas, evitar a identificação do paciente.

O método computacional projetado será uma Rede Neural Convolutacional (CNN), atualmente muito utilizado no processamento e classificação de imagens digitais. Com auxílio do API (Interface de Programação de Aplicação) Keras, será feita, utilizando a linguagem de programação Python, um treino de diferentes modelos de CNN, para definir aquele que é mais preciso e sensível na identificação de lesões causadas pela Mpox. Os modelos selecionados para serem estudados neste projeto serão VGG-16, VGG-19, Xception, ResNet50, MobileNet\_V2, Inception\_V3 e DenseNet\_201

O conceito de Cross-entropy Loss (Perda de Entropia Cruzada) consiste em uma função de perda, ou seja, uma função para quantificar o quão próximo do correto estão os resultados obtidos pelo modelo. A Cross-entropy Loss é utilizada para este estudo pois permite a análise da perda de mais de dois rótulos, e tem como resultado um valor da perda total resultante de todos os outputs gerados pelo modelo.

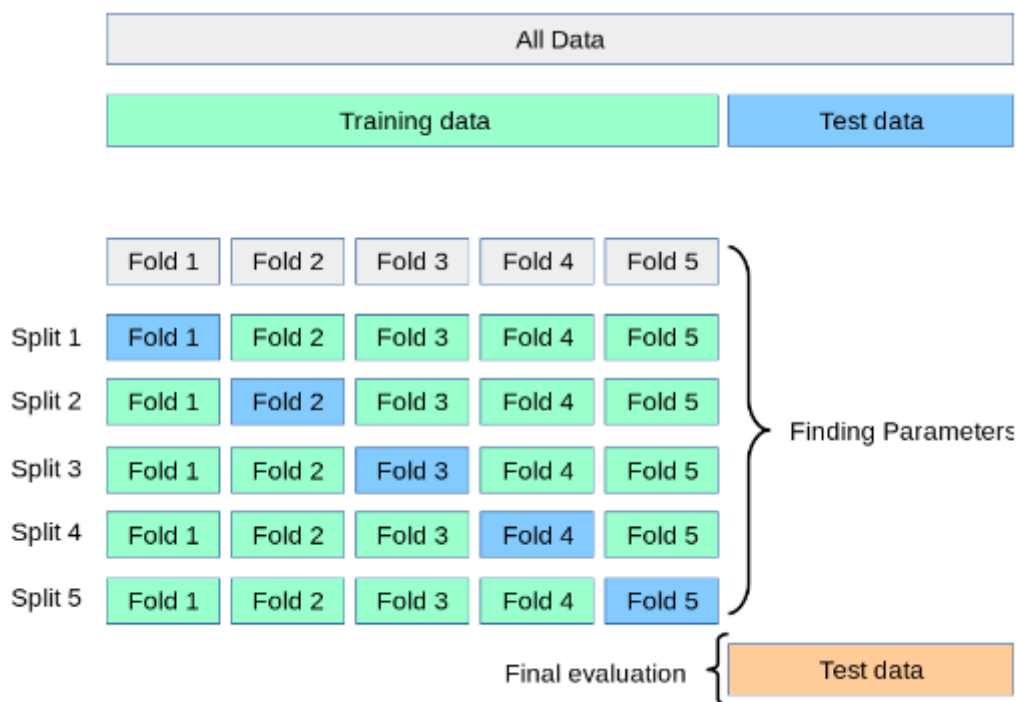
Quanto maior a perda obtida pela função, maior será a diferença entre as probabilidades obtidas e as probabilidades reais, o que indica ao modelo que os parâmetros devem se alterar. Portanto, é desejável que o modelo apresente o menor valor de perda possível.

As imagens previamente rotuladas serão utilizadas como *input* nos diferentes modelos de CNN que serão testados, assim regulando os parâmetros da rede neural, gerando um modelo treinado, que resulte nos *outputs* mais próximos do indicado em cada imagem de treinamento (Figura 1.B).

Os modelos serão treinados a partir do código previamente construído pela equipe do CSBL.

Para garantir que o modelo treinado seja capaz de prever corretamente as doenças em novas imagens, não presentes no *Dataset*, também será realizada a validação do modelo. Devido o tamanho limitado do *Dataset* utilizado, com apenas 701 imagens, será utilizada a técnica de validação cruzada 5-fold (5-fold cross-validation), que consiste na divisão das imagens rotuladas em 5 partes de tamanho similar, denominadas *folds*, o modelo será treinado 5 vezes, em cada uma, um *fold* diferente será utilizado para validação, enquanto os outros serão usados como conjunto de treinamento. Por fim, é feita a média dos resultados obtidos para se obter o desempenho do modelo (SCIKIT-LEARN, 2023)

**Fig.2** Esquema da Validação Cruzada 5-Fold, indicando as divisões do *dataset*



**FONTE:** Scikit-Learn, 2023

### 3.1. ANÁLISE ESTATÍSTICA

Após o modelo ser treinado, serão realizados testes com as imagens não rotuladas, que proporcionarão resultados, que podem ser analisados estatisticamente (Figura 1.C). As métricas estatísticas que serão utilizadas para avaliar os diferentes modelos serão a acurácia, precisão, recall (sensibilidade) e F1-Score (BIOINFO, 2021).

Para a análise binária, os resultados dos modelos serão divididos 4 resultados possíveis:

- Casos rotulados como Mpox identificados pelo modelo como sendo Mpox (Verdadeiros Positivos)
- Casos rotulados como não Mpox identificados pelo modelo como não sendo Mpox (Verdadeiros Negativos)

- Casos rotulados como não Mpox identificados pelo modelo como sendo Mpox (Falsos Positivos)
- Casos rotulados como Mpox identificados pelo modelo como não sendo Mpox (Falsos Negativos)

**Tabela 2** - Matriz de Confusão Binária

Predito	Rótulo	
	Mpox	Não Mpox
Mpox	Verdadeiro Positivo (VP)	Falso Positivo (FP)
Não Mpox	Falso Negativo (FN)	Verdadeiro Negativo (VN)

FONTE: Elaborado pelo autor

Está sendo analisado 6 classes (Herpes Simples, Impetigo, Molluscum Contagiosum, Mpox, Syphilis, Varicella), portanto, também será realizada a análise *One versus Rest (OvR)* dos parâmetros, para classificação multiclasse. Para isso será utilizada a biblioteca de código aberto *Scikit-Learn*, que disponibiliza uma variedade de ferramentas para calcular estes parâmetros.

Os códigos em *Python* utilizado para as análises pode ser encontrado na seção Apêndice.

### 3.1.1 ACURÁCIA

A acurácia avalia a performance do modelo com um todo, relacionando o número de resultados corretos (verdadeiros positivos e verdadeiros negativos) com o número total de resultados (verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos)

$$A = \frac{VP + VN}{VP + VN + FP + FN}$$

### 3.1.2 PRECISÃO

A precisão é a medida da habilidade do modelo em prever corretamente os resultados verdadeiros positivos, em relação a todos os resultados previstos como positivos (verdadeiros positivos e falsos positivos)

$$P = \frac{VP}{VP + FP}$$

### 3.1.3 RECALL (SENSIBILIDADE)

O recall representa a capacidade do modelo em identificar corretamente os resultados verdadeiros positivos em relação a todos os resultados verdadeiros existentes (verdadeiros positivos e falsos negativos).

$$R = \frac{VP}{VP + FN}$$

### 3.1.4 F1-SCORES

O F1-Score é a média harmônica entre a precisão (P) e o recall (R), sendo uma métrica que representa ambos esses valores.

$$F_1 Score = \frac{2 * P * R}{P + R}$$

## 4. RESULTADOS

Primeiramente foram gerados os resultados dos modelos utilizando a mesma parte do *Dataset* que foi utilizada para o treinamento, obtendo-se as seguintes matrizes de confusão:

**Tabela 3** - Relação entre resultado predito do treinamento através do modelo “VGG-16” e resultado esperado

VGG-16						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	59,60%	19,20%	1,00%	0,00%	20,20%	0,00%
Impetigo	2,80%	87,90%	2,80%	0,90%	4,70%	0,90%
Molluscum	0,80%	6,40%	86,40%	0,80%	4,00%	0,90%
Monkeypox	0,00%	0,00%	0,00%	94,30%	5,70%	0,00%
Syphilis	1,50%	3,00%	1,50%	0,00%	93,90%	0,00%
Varicella	1,70%	2,60%	6,00%	0,00%	6,00%	83,60%

**FONTE:** Elaborado pelo autor

**Tabela 4** - Relação entre resultado predito do treinamento através do modelo “VGG-19” e resultado esperado

VGG-19						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	74,70%	12,10%	8,10%	0,00%	5,10%	0,00%
Impetigo	2,80%	86,00%	6,50%	0,00%	1,90%	2,80%
Molluscum	1,60%	4,80%	88,00%	0,80%	0,80%	4,00%
Monkeypox	1,90%	1,90%	1,90%	90,60%	3,80%	0,00%
Syphilis	5,60%	5,60%	3,00%	0,00%	85,30%	0,50%
Varicella	0,00%	3,40%	8,60%	0,90%	1,70%	85,30%

**FONTE:** Elaborado pelo autor

**Tabela 5** - Relação entre resultado predito do treinamento através do modelo “Xception” e resultado esperado

Xception						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella



	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	70,70%	9,10%	4,00%	2,00%	12,10%	2,00%
Impetigo	27,10%	62,60%	0,90%	0,90%	2,80%	5,60%
Molluscum	4,80%	38,40%	44,80%	1,60%	5,60%	4,80%
Monkeypox	7,50%	24,50%	11,30%	49,10%	7,50%	0,00%
Syphilis	8,10%	18,30%	3,00%	0,50%	67,50%	2,50%
Varicella	0,90%	15,50%	13,80%	0,90%	6,00%	62,90%

**FONTE:** Elaborado pelo autor

**Tabela 6** - Relação entre resultado predito do treinamento através do modelo “ResNet50” e resultado esperado

ResNet50						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	77,80%	8,10%	6,10%	0,00%	8,10%	0,00%
Impetigo	0,90%	93,50%	0,90%	0,90%	0,00%	3,70%
Molluscum	0,00%	2,40%	93,60%	0,00%	2,40%	1,60%
Monkeypox	0,00%	0,00%	0,00%	94,30%	5,70%	0,00%
Syphilis	2,00%	0,50%	4,10%	0,50%	92,90%	0,00%
Varicella	0,90%	3,40%	3,40%	1,70%	0,90%	89,70%

**FONTE:** Elaborado pelo autor

**Tabela 7** - Relação entre resultado predito do treinamento através do modelo “MobileNet\_V2” e resultado esperado

MobileNet_V2						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	66,70%	15,20%	7,10%	0,00%	11,10%	0,00%
Impetigo	8,40%	79,40%	2,80%	0,90%	4,70%	3,70%

Molluscum	2,40%	4,80%	79,20%	0,80%	4,80%	8,00%
Monkeypox	9,40%	3,80%	1,90%	69,80%	9,40%	5,70%
Syphilis	8,60%	3,60%	4,10%	1,50%	81,20%	1,00%
Varicella	0,90%	2,60%	6,00%	0,90%	8,60%	81,00%

**FONTE:** Elaborado pelo autor

**Tabela 8** - Relação entre resultado predito do treinamento através do modelo “Inception\_V3” e resultado esperado

Inception_V3						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	63,60%	5,10%	8,10%	0,00%	23,20%	0,00%
Impetigo	3,70%	71,00%	6,50%	2,80%	14,00%	1,90%
Molluscum	3,20%	4,80%	83,20%	0,00%	3,20%	5,60%
Monkeypox	1,90%	5,70%	3,80%	75,50%	11,30%	1,90%
Syphilis	2,50%	2,00%	4,10%	1,00%	87,80%	2,50%
Varicella	0,00%	3,40%	6,90%	0,90%	5,20%	83,60%

**FONTE:** Elaborado pelo autor

**Tabela 9** - Relação entre resultado predito do treinamento através do modelo “DenseNet\_201” e resultado esperado

DenseNet_201						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	84,80%	2,00%	1,00%	0,00%	11,10%	1,00%
Impetigo	26,20%	57,00%	3,70%	1,90%	8,40%	2,80%
Molluscum	10,40%	9,60%	66,40%	1,60%	9,60%	2,40%
Monkeypox	9,40%	3,80%	1,90%	84,90%	0,00%	0,00%
Syphilis	12,70%	1,50%	4,10%	4,10%	77,20%	0,50%

Varicella	0,00%	3,40%	8,60%	1,70%	8,60%	77,60%
-----------	-------	-------	-------	-------	-------	--------

**FONTE:** Elaborado pelo autor

Posteriormente foi realizada a validação dos modelos.

Para cada resultado gerado pelos modelos, o rótulo predito é aquele que apresentou maior probabilidade.

Relacionando os rótulos preditos (Predição) e os rótulos verdadeiros (Rótulo), foram obtidos os seguintes resultados:

**Tabela 10** - Relação entre resultado predito pela validação através do modelo “VGG-16” e resultado esperado

VGG-16						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	12,10%	13,16%	14,14%	8,09%	42,43%	10,06%
Impetigo	14,98%	11,21%	19,61%	4,69%	33,62%	15,90%
Molluscum	18,40%	11,99%	21,58%	4,80%	31,23%	11,99%
Monkeypox	9,46%	9,46%	32,06%	13,14%	24,57%	11,30%
Syphilis	12,70%	18,26%	18,79%	3,54%	27,42%	19,29%
Varicella	16,41%	10,34%	23,26%	8,59%	24,16%	17,25%

**FONTE:** Elaborado pelo autor

**Tabela 11** - Relação entre resultado predito pela validação através do modelo “VGG-19” e resultado esperado

VGG-19						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	19,23%	18,17%	20,21%	10,07%	20,21%	12,11%
Impetigo	12,17%	14,00%	21,48%	5,60%	31,77%	14,97%
Molluscum	14,38%	12,82%	21,57%	4,01%	31,22%	16,00%
Monkeypox	15,13%	18,82%	28,29%	9,47%	18,82%	9,47%
Syphilis	13,19%	16,24%	16,24%	4,07%	30,46%	19,81%
Varicella	8,60%	15,51%	14,67%	5,17%	37,94%	18,10%

**FONTE:** Elaborado pelo autor

**Tabela 12** - Relação entre resultado predito pela validação através do modelo “Xception” e resultado esperado

Predição	Xception					
	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	20,20%	14,14%	14,14%	5,07%	31,32%	15,13%
Impetigo	15,90%	9,32%	14,98%	1,89%	33,62%	24,30%
Molluscum	16,80%	7,98%	18,42%	4,80%	35,21%	16,80%
Monkeypox	13,16%	16,97%	22,63%	3,82%	35,92%	7,50%
Syphilis	17,27%	16,74%	16,24%	5,06%	27,42%	17,27%
Varicella	24,16%	20,67%	15,50%	5,17%	20,67%	13,82%

**FONTE:** Elaborado pelo autor

**Tabela 13** - Relação entre resultado predito pela validação através do modelo “ResNet50” e resultado esperado

Predição	ResNet50					
	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	25,26%	14,14%	10,06%	5,07%	25,26%	20,20%
Impetigo	16,81%	14,98%	10,29%	5,60%	33,62%	18,70%
Molluscum	16,00%	16,00%	14,38%	6,41%	33,61%	13,60%
Monkeypox	11,32%	18,82%	20,79%	13,16%	26,45%	9,47%
Syphilis	8,14%	22,34%	16,74%	2,02%	28,42%	22,34%
Varicella	6,01%	25,86%	18,10%	8,60%	22,43%	19,00%

**FONTE:** Elaborado pelo autor

**Tabela 14** - Relação entre resultado predito pela validação através do modelo “MobileNet\_V2” e resultado esperado

MobileNet_V2						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	28,29%	13,16%	16,19%	4,01%	35,33%	3,03%
Impetigo	16,81%	9,32%	23,39%	3,71%	39,28%	7,49%
Molluscum	12,82%	17,61%	24,80%	3,18%	34,39%	7,19%
Monkeypox	13,18%	22,66%	18,84%	11,33%	26,48%	7,51%
Syphilis	15,22%	23,36%	16,74%	3,54%	34,02%	7,12%
Varicella	19,83%	14,66%	26,74%	3,43%	26,74%	8,59%

**FONTE:** Elaborado pelo autor

**Tabela 15** - Relação entre resultado predito pela validação através do modelo “Inception\_V3” e resultado esperado

Inception_V3						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	24,23%	17,18%	10,07%	5,07%	26,27%	17,18%
Impetigo	12,17%	6,51%	21,48%	4,69%	40,17%	14,97%
Molluscum	13,61%	9,59%	20,80%	4,02%	37,59%	14,39%
Monkeypox	9,47%	13,16%	16,97%	11,32%	37,76%	11,32%
Syphilis	15,22%	12,70%	19,29%	3,54%	30,47%	18,79%
Varicella	9,50%	16,41%	20,67%	6,91%	31,01%	15,50%

**FONTE:** Elaborado pelo autor

**Tabela 16** - Relação entre resultado predito pelo modelo “DenseNet\_201” e resultado esperado

DenseNet_201						
Predição	Rótulo					
	Herpes	Impetigo	Molluscum	Monkeypox	Syphilis	Varicella
Herpes	12,11%	18,17%	15,14%	5,07%	38,38%	11,13%
Impetigo	10,29%	14,01%	20,59%	4,69%	33,62%	16,81%
Molluscum	13,60%	14,38%	20,01%	7,19%	29,60%	15,22%
Monkeypox	9,46%	16,95%	26,41%	9,46%	24,57%	13,14%
Syphilis	12,70%	19,29%	14,72%	2,55%	32,48%	18,26%
Varicella	12,92%	24,16%	14,66%	6,91%	28,43%	12,92%

**FONTE:** Elaborado pelo autor

A partir destes resultados, a primeira análise feita para analisar a validação foi de forma binária:

**Tabela 17** - Métricas de desempenho da validação dos diferentes modelos, análise binária

Modelo	Acurácia	Precisão	Recall	F1-Score
VGG-16	5,16%	16,28%	13,21%	14,58%
VGG-19	5,02%	12,50%	9,43%	10,75%
Xception	4,16%	6,45%	3,77%	4,76%
ResNet50	4,73%	17,50%	13,21%	15,05%
MobileNet_V2	3,30%	20,69%	11,32%	14,64%
Inception_V3	4,30%	16,67%	11,32%	13,48%
DenseNet_201	4,59%	13,51%	9,43%	11,11%

**FONTE:** Elaborado pelo autor

Utilizando os dados que relacionam os rótulos preditos e os rótulos verdadeiros, foi feita a análise multiclasse OvR, obtendo os seguintes resultados:

**Tabela 18** - Métricas de desempenho da validação dos diferentes modelos, análise multiclasse *one vs rest*

Modelo	Acurácia	Precisão	Recall	F1-Score
VGG-16	19,51%	17,95%	17,71%	17,38%
VGG-19	21,00%	16,24%	16,11%	16,14%
Xception	17,93%	15,32%	15,48%	15,29%
ResNet50	20,54%	19,76%	19,08%	19,31%
MobileNet_V2	21,69%	20,20%	19,26%	19,00%
Inception_V3	20,22%	18,44%	18,14%	18,16%
DenseNet_201	19,51%	17,18%	16,83%	16,90%

**FONTE:** Elaborado pelo autor

## 5. DISCUSSÃO

Apesar das matrizes de confusão dos treinamentos apresentarem ótimos valores de acurácia, os resultados obtidos na validação, tanto na análise binária quanto na análise OvR, demonstraram um desempenho significativamente abaixo do ideal, tornando inviável a utilização do código construído em um contexto prático. Essa limitação pode ser atribuída, em grande parte, à inadequação do *Dataset* utilizado para treinar o modelo. Este conjunto de dados se mostrou inadequado devido a sua escassez de imagens, pois se trata de um problema complexo, o que necessitaria do modelo identificar padrões e nuances em um grande número de lesões para uma diferenciação mais precisa das doenças. Em outros estudos, como o de Shams Nafisa (2022), que buscou apenas diferenciar a Mpox, o número de imagens utilizadas era similar, contudo foi feito a *Data Augmentation*, método para aumento do *Dataset* de treinamento, com isso, os resultados apresentados foram muito mais satisfatórios, com uma acurácia mais de 80% na validação do modelo.

Além disso, o conjunto de dados apresenta um desequilíbrio notável no número de imagens associadas a cada classe de doença. Algumas doenças são representadas por uma quantidade muito maior de imagens em comparação com outras. Por exemplo, a classe da Syphilis abrange quase 30% de todas as amostras, enquanto a classe de Mpox não alcança sequer 10%. Esse desequilíbrio tem o efeito de induzir um viés no modelo, levando-o a favorecer a rotulação das

imagens em relação às classes majoritárias, resultando em resultados desequilibrados e prejudicando a capacidade do modelo de generalização.

A abordagem de *ensemble* dos modelos, que consiste na junção dos diferentes modelos em um único resultado, também poderia ter sido utilizado para produzir melhores resultados, assim como feito no trabalho de Chiranjibi e Tej Bahadur (2022), onde todos os parâmetros apresentaram mais de 85%. Superando todos os modelos testados isoladamente.

Portanto, a aquisição de um volume maior de imagens, bem como a correção do desequilíbrio entre as classes e a implementação de outras estratégias na construção artificial de mais imagens (*Data Augmentation*), são necessárias para melhorar o desempenho e a utilidade do modelo construído. Com essas medidas, será possível explorar todo o potencial do modelo em um ambiente real e alcançar resultados capazes de serem utilizados para o diagnóstico diferencial da Mpox em relação a outras lesões de pele.

## 7. CONCLUSÕES

Neste trabalho buscamos desenvolver um método computacional que pudesse auxiliar profissionais da medicina no reconhecimento de lesões de pele para diagnóstico diferencial de pacientes com a Varíola do Macaco, a fim de agilizar a conclusão do diagnóstico. Contudo, é necessário que sejam realizadas algumas das técnicas utilizadas em outros trabalhos, como por exemplo o uso de *Data Augmentation*, para aumentar o número de imagens para treinar o modelo. Por isso é de suma importância que futuramente novos estudos sejam realizados, analisando diferentes modelos e melhorando o código.

Entretanto, mesmo obtendo resultados insatisfatórios, este estudo demonstra a relevância de utilizar abordagens tecnológicas para desenvolvimento do campo da medicina, de forma de promover o bem estar dos pacientes e prevenir potenciais próximas pandemias globais.

## 7. REFERÊNCIAS



**WHO Director-General declares monkeypox outbreak a public health emergency of international concern - PAHO/WHO | Pan American Health Organization.** Disponível em:

<<https://www.paho.org/en/news/23-7-2022-who-director-general-declares-monkeypox-outbreak-public-health-emergency>>. Acesso em: 29 ago. 2023.

SKLENOVSKÁ, N.; VAN RANST, M. Emergence of Monkeypox as the Most Important Orthopoxvirus Infection in Humans. **Frontiers in Public Health**, v. 6, 2018a.

SKLENOVSKÁ, N.; VAN RANST, M. Emergence of Monkeypox as the Most Important Orthopoxvirus Infection in Humans. **Frontiers in Public Health**, v. 6, 2018b.

CDC. Mpox in the U.S. Disponível em: <<https://www.cdc.gov/poxvirus/mpox/response/2022/index.html>>.

KALER, J. et al. Monkeypox: A Comprehensive Review of Transmission, Pathogenesis, and Manifestation. *Cureus*, v. 14, n. 7, 3 jul. 2022.

KOZLOV, M. The monkeypox virus is mutating. Are scientists worried? **Nature**, 5 out. 2022b.

KOZLOV, M. Can a smallpox drug treat monkeypox? Here's what scientists know. **Nature**, v. 609, n. 7927, p. 451–452, 25 ago. 2022a.

HRAIB, M. et al. The outbreak of monkeypox 2022: An overview. **Annals of Medicine and Surgery**, v. 79, p. 104069, 1 jul. 2022.

SHEHAB, M. et al. Machine learning in medical applications: A review of state-of-the-art methods. **Computers in Biology and Medicine**, v. 145, p. 105458, 1 jun. 2022.

DOSHI-VELEZ, F.; KIM, B. **Towards A Rigorous Science of Interpretable Machine Learning**, 2017. Disponível em: <<https://arxiv.org/pdf/1702.08608.pdf>>.

BURKART, N.; HUBER, M. F. **A Survey on the Explainability of Supervised Machine Learning**. *Journal of Artificial Intelligence Research*, v. 70, p. 245–317, 19 jan. 2021.

VELLIDO, A. The importance of interpretability and visualization in machine learning for applications in medicine and health care. **Neural Computing and Applications**, 4 fev. 2019.

**Home | Hellenic Dermatological Atlas - Over 2700 Dermatology pictures**. Disponível em: <<http://www.hellenicdermatlas.com/en/>>. Acesso em: 24 jan. 2023.  
HRAIB, M. et al. The outbreak of monkeypox 2022: An overview. **Annals of Medicine and Surgery**, v. 79, p. 104069, 1 jul. 2022.

**Keras documentation: Keras Applications**. Disponível em: <<https://keras.io/api/applications/>>.

SCIKIT-LEARN. **sklearn.metrics.log\_loss**. Disponível em: <[https://scikit-learn.org/stable/modules/generated/sklearn.metrics.log\\_loss.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.log_loss.html)>.

SCIKIT-LEARN. **3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.21.3 documentation**. Disponível em: <[https://scikit-learn.org/stable/modules/cross\\_validation.html](https://scikit-learn.org/stable/modules/cross_validation.html)>.

**Métricas de avaliação em machine learning | BIOINFO**. Disponível em: <<https://bioinfo.com.br/metricas-de-avaliacao-em-machine-learning-acuracia-sensibilidade-precisao-especificidade-e-f-score/>>. Acesso em: 3 fev. 2023.


**Ilustração de uma curva ROC. Adaptado de MartinThoma (Creative Commons CC0 1.0 Universal Public Domain Dedication)**. Disponível em: <[https://commons.wikimedia.org/wiki/File:Curva\\_ROC.svg](https://commons.wikimedia.org/wiki/File:Curva_ROC.svg)>. Acesso em: 5 set. 2023.


ALI, S. et al. **Monkeypox Skin Lesion Detection Using Deep Learning Models: A Feasibility Study**. [s.l: s.n.]. Disponível em: <<https://arxiv.org/pdf/2207.03342.pdf>>.

SITAULA, C.; SHAHI, T. B. **Monkeypox Virus Detection Using Pre-trained Deep Learning-based Approaches**. Journal of Medical Systems, v. 46, n. 11, 6 out. 2022.

BALA, D. et al. **MonkeyNet: A robust deep convolutional neural network for monkeypox disease detection and classification**. Neural Networks, v. 161, p. 757–775, 1 abr. 2023.

São Paulo, 9 de Outubro de 2023.

  
João Gabriel Nascimento Castilho

  
Prof. Dr. Mauro César Cafundó de Moraes

## 8. APÊNDICES

### APÊNDICE A - Código utilizado para obtenção dos resultados do modelo no *Dataset* de treino

```
# %%  
# Imports  
import os  
import numpy as np  
import keras  
  
import matplotlib.pyplot as plt  
from sklearn.metrics import confusion_matrix, classification_report  
from sklearn.metrics import ConfusionMatrixDisplay  
from load_data import load_dataset  
  
# %%  
# Load the trained model  
model_path = "C:\\Users\\jgnca\\Downloads\\drive-download-20230925T220732Z-001\\results\\models\\Xception_data_augmentation.keras"  
model = keras.models.load_model(model_path)  
print(model.summary())  
  
# %%  
# Load the training dataset  
TRAIN_DIR = "C:\\Users\\jgnca\\Downloads\\drive-download-20230925T220732Z-001\\data\\train"  
  
image_dataset, labels_dataset = load_dataset(  
    dataset_dir = TRAIN_DIR,  
    image_size = (72, 72),  
    shuffle=False, random_seed=42  
)  
  
# %%  
# Estimate predictions  
y_pred = model.predict(image_dataset)  
  
# Convert predicted probabilities to class labels
```

```

y_pred_classes = np.argmax(y_pred, axis=1)

print(y_pred_classes)

# %%
# Calculate the Confusion Matrix
conf_matrix = confusion_matrix(labels_dataset, y_pred_classes)
row_sums = conf_matrix.sum(axis=1, keepdims=True)
# It's important to normalize the
# confusion matrix by dividing each value by the total number of images
# in the
# corresponding (true) class (i.e., divide by the row's sum).
norm_conf_matrix = conf_matrix / row_sums

# %%
# Print Classification Report
class_names = ['herpes', 'impetigo', 'molluscum', 'monkeypox', 'syphi-
lis', 'varicella']
print(classification_report(labels_dataset, y_pred_classes, tar-
get_names=class_names))

# %%
# Plot confusion matrix and error analysis
sample_weight = (y_pred_classes != labels_dataset)
ConfusionMatrixDisplay.from_predictions(
    labels_dataset, y_pred_classes,
    #sample_weight=sample_weight, # uncomment this line to see the er-
    #ror analysis matrix
    normalize="true", values_format=".1%",
    cmap="Blues")
tick_marks = np.arange(len(class_names))
plt.xticks(tick_marks, class_names, rotation=45)
plt.yticks(tick_marks, class_names)
plt.show()

# %%
# Misclassified images
cl_a, cl_b = 0, 4 # herpes and syphilis
X_aa = image_dataset[(labels_dataset == cl_a) & (y_pred_classes ==
cl_a), ...]

```

```

X_ab = image_dataset[(labels_dataset == cl_a) & (y_pred_classes ==
cl_b), ...]
X_ba = image_dataset[(labels_dataset == cl_b) & (y_pred_classes ==
cl_a), ...]
X_bb = image_dataset[(labels_dataset == cl_b) & (y_pred_classes ==
cl_b), ...]

# %%
# Plot Misclassified Images
size = 3
pad = 0.2
plt.figure(figsize=(6, 6))
for images, (label_col, label_row) in [(X_ba, (0, 0)), (X_bb, (1, 0)),
(X_aa, (0, 1)), (X_ab, (1, 1))]:
    for idx, image_data in enumerate(images[:size*size]):
        x = idx % size + label_col * (size + pad)
        y = idx // size + label_row * (size + pad)
        plt.imshow(image_data, #.reshape(64, 64), cmap="Grays",
                    extent=(x, x + 1, y, y + 1))
plt.xticks([size / 2, size + pad + size / 2], ["herpes", "syphilis"])
plt.yticks([size / 2, size + pad + size / 2], ["syphilis", "herpes"])
plt.plot([size + pad / 2, size + pad / 2], [0, 2 * size + pad], "k:")
plt.plot([0, 2 * size + pad], [size + pad / 2, size + pad / 2], "k:")
plt.axis([0, 2 * size + pad, 0, 2 * size + pad])
plt.xlabel("Predicted label")
plt.ylabel("True label")
#save_fig("error_analysis_digits_plot")
plt.show()

# %%
# TODO: Binary prediction
threshold = 0.5
binary_prediction = np.where(y_pred > threshold, 1, 0)

```

Fonte: CSBL - USP

## APÊNDICE B - Código utilizado para obtenção dos resultados do modelo no *Dataset* de validação

```
import pandas as pd
```

```

from sklearn.metrics import confusion_matrix, accuracy_score, precision_score, recall_score, f1_score, classification_report
from sklearn.preprocessing import LabelEncoder

# Carregar dados
df = pd.read_excel('Xception_Dados.xlsx')

# Reconhecer colunas do excel e numerar
verdadeiros = df['verdadeiros']
preditos = df['preditos']

label_encoder = LabelEncoder()
verdadeiros_encoded = label_encoder.fit_transform(verdadeiros)
preditos_encoded = label_encoder.transform(preditos)

# Calcular a matriz
matriz_confusao = confusion_matrix(verdadeiros_encoded,
preditos_encoded)

# Calcular a acurácia multiclasse, a precisão, recall e F1-score para
todas as classes
acuracia = accuracy_score(verdadeiros_encoded, preditos_encoded)
precisao = precision_score(verdadeiros_encoded, preditos_encoded,
average='macro')
recall = recall_score(verdadeiros_encoded, preditos_encoded,
average='macro')
f1 = f1_score(verdadeiros_encoded, preditos_encoded, average='macro')

# Imprimir a matriz
print("Matriz de Confusão:")
print(matriz_confusao)

# Imprimir relatório
print("Relatório de Classificação:")
print(classification_report(verdadeiros_encoded, preditos_encoded,
target_names=df['verdadeiros'].unique()))

# Imprimir as métricas
print('Acurácia:', acuracia)
print('Precisão:', precisao)
print('Recall:', recall)

```

```
print('F1-score:', f1)
```

**FONTE:** Elaborado pelo autor