

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**MARCELO SCHREIBER FERNANDES**

**IDENTIFICAÇÃO DO MOSQUITO *Aedes Aegypti* ATRAVÉS DE ÁUDIO  
UTILIZANDO TÉCNICAS DE *MACHINE LEARNING***

Porto Alegre  
2018

MARCELO SCHREIBER FERNANDES

**IDENTIFICAÇÃO DO MOSQUITO *Aedes Aegypti* ATRAVÉS DE ÁUDIO  
UTILIZANDO TÉCNICAS DE *MACHINE LEARNING***

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica na Universidade Federal do Rio Grande do Sul, como parte dos requisitos para a obtenção do título de Engenheiro Eletricista.

Orientador: Prof.<sup>a</sup> Dra. Mariana Recamonde Mendoza  
Coorientador: Prof. Dr. Weverton Luis da Costa Cordeiro

Porto Alegre  
2018

MARCELO SCHREIBER FERNANDES

**IDENTIFICAÇÃO DO MOSQUITO *Aedes Aegypti* ATRAVÉS DE ÁUDIO  
UTILIZANDO TÉCNICAS DE *MACHINE LEARNING***

Este projeto foi julgado adequado para fazer jus aos créditos da Disciplina de “Projeto de Diplomação”, do Departamento de Engenharia Elétrica e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: \_\_\_\_\_

Prof.<sup>a</sup> Dra. Mariana Recamonde Mendoza

Doutora pela Universidade Federal do Rio Grande do Sul  
– Porto Alegre, Brasil

Aprovado em: \_\_\_\_/\_\_\_\_/\_\_\_\_

Banca Examinadora:

Prof. Dr. Bruno Castro da Silva, UFRGS  
Doutor pela University of Massachusetts – Amherst, Estados Unidos

Prof. Dr. Ivan Müller, UFRGS  
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Prof. Dr. Roger Flores Ceccon, UFRGS  
Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Porto Alegre, dezembro de 2018.

## RESUMO

Mosquitos são vetores de doenças infecciosas, tais como malária, dengue e febre amarela, e afetam milhares de pessoas anualmente. Uma maneira de combater esses vetores é realizando o mapeamento geográfico de espécies. O presente trabalho explora a possibilidade de identificar o mosquito *Aedes aegypti*, em específico, utilizando seu áudio e técnicas de *Machine Learning*. De forma que trabalhos futuros possam utilizar o modelo de classificador proposto por este trabalho para auxiliar no mapeamento de espécies de mosquito. Cada espécie apresenta uma frequência de voo diferente. Diferentes estudos apontam diferentes faixas de frequência, em geral, a mesma está compreendida entre 200 a 700 Hz. Usou-se um *dataset* contendo gravações realizadas com aparelhos celulares do voo de 20 espécies, sendo duas dessas espécies subclassificadas em estirpes, totalizando 23 classes diferentes. Realizou-se a reamostragem dessas gravações para a frequência de 8 kHz e removeu-se trechos onde não havia som de mosquitos voando. Treinou-se redes neurais convolucionais utilizando aprendizado supervisionado e validação cruzada *K-fold* com 10-*folds*. Como *feature* usou-se o espectrograma das gravações, de forma que a frequência de voo do mosquito seja representada visualmente ao longo do tempo. Foram treinados três classificadores com o objetivo de comparar os resultados dos mesmos; classificador binário entre *Aedes aegypti* vs. restante das classes, classificador multiclasse e um *ensemble* composto de classificadores binários (onde cada classificador binário realiza a classificação entre *Aedes aegypti* vs. uma espécie específica). Obtiveram-se métricas de desempenho superiores aos trabalhos encontrados na literatura, os quais não usam *Machine Learning* para realizar a classificação. O classificador multiclasse apresentou acurácia de  $80,00\% \pm 1,3$ , já o classificador binário *Aedes aegypti* vs. demais classes teve acurácia de  $97,78\% \pm 0,73$  e, por fim, o *ensemble* de classificadores binários apresentou o melhor resultado com acurácia de  $98,34\% \pm 0,27$ . Portanto, conclui-se ser possível realizar a identificação de espécies de mosquitos, em especial do *Aedes aegypti*, utilizando o som de seu voo e técnicas de *Machine Learning*.

Palavras-chave: *Machine Learning*, mosquito, *Aedes aegypti*, redes neurais convolucionais.

## ABSTRACT

Mosquitos are a vector of infectious diseases, such as malaria, dengue and yellow fever, and affect thousands of people annually. One way to combat these vectors is by mapping these species geographically. This work explores the possibility of identifying the *Aedes aegypti* mosquito in specific, using its audio and Machine Learning techniques. So that future works can use the classifier model proposed here to assist in the mapping of mosquito species. Each species has a different flight frequency, different studies show different frequency ranges, and usually, it's contained between 200 and 700 Hz. It was used a dataset containing cellphone flight recordings of 20 species, two of these species being subclassified into strains, totaling 23 different classes. All recordings were resampled to 8 kHz and segments that didn't have a sound of mosquitoes flying were removed. Convolutional neural networks were trained using supervised learning and K-fold cross-validation with 10-folds. The recordings' spectrograms were used as a feature so that the flight frequency of the mosquito is represented visually along the time. Three classifiers were trained to compare their results: binary classifier between *Aedes aegypti* vs remainder classes, multiclass classifier and an ensemble composed by binary classifiers (where each binary classifier performs the classification between *Aedes aegypti* vs a specific species). The performance metrics obtained were higher than those found in other works, which do not use Machine Learning to perform the classification. The multiclass classifier had an accuracy of  $80.00\% \pm 1.3$ , whereas the binary classifier *Aedes aegypti* vs remainder classes had an accuracy of  $97.78\% \pm 0.73$  and the ensemble shown the best result with an accuracy of  $98.34 \pm 0.27$ . Therefore, it is possible to identify mosquito species, especially *Aedes aegypti*, using their flight sound and Machine Learning techniques.

Keywords: Machine Learning, mosquitos, *Aedes aegypti*, convolutional neural networks.

## LISTA DE FIGURAS

Figura 1 - Exemplo do espectrograma e distribuição de frequência de um mosquito.....	13
Figura 2 - Equivalência entre escala Hertz e mel .....	14
Figura 3 - Representação do tamanho de janela e <i>hop length</i> da transformada de Fourier.....	15
Figura 4 – Comparação entre paradigmas de programação .....	16
Figura 5 - Representação de um neurônio de uma rede neural .....	17
Figura 6 - Rede neural multiclasse contendo duas camadas ocultas .....	18
Figura 7 - Exemplo da operação de convolução.....	20
Figura 8 - Operação de <i>pooling</i> em uma imagem 4×4 usando um filtro 2×2 com <i>stride</i> 2 .....	20
Figura 9 - Função de ativação sigmóide.....	21
Figura 10 - Função de ativação ReLU .....	22
Figura 11 - Exemplo de validação cruzada <i>k-fold</i> com <i>k</i> =3 .....	23
Figura 12 - Matriz de confusão de um classificador binário .....	23
Figura 13 - Fluxograma da metodologia .....	29
Figura 14 - Histograma da duração .....	31
Figura 15 - Histograma da frequência de amostragem.....	31
Figura 16- Espectrograma do <i>Aedes aegypti</i> .....	33
Figura 17 - Espectrograma do <i>Aedes albopictus</i> .....	33
Figura 18 - Espectrograma do <i>Aedes mediovittatus</i> .....	33
Figura 19 - Espectrograma do <i>Culex pipiens</i> .....	34
Figura 20 - Espectrograma do <i>Culex quinquefasciatus</i> .....	34
Figura 21 - Delta do espectrograma do <i>Aedes aegypti</i> .....	34
Figura 22 - Transformada de Fourier com 3 bandas e 3 quadros .....	37
Figura 23 - Representação em decibéis da transformada de Fourier.....	37
Figura 24 - Transformada de Fourier após normalização.....	38
Figura 25 - Modelo de CNN com o melhor desempenho, nesse caso por ter 2 neurônios na saída é um classificador binário .....	41
Figura 26 - Acurácia do classificador <i>Aedes aegypti</i> vs. <i>Anopheles freeborni</i> variando-se os parâmetros .....	43
Figura 27 - Precisão do classificador <i>Aedes aegypti</i> vs. <i>Anopheles freeborni</i> variando-se os parâmetros .....	43
Figura 28 - <i>Recall</i> do classificador <i>Aedes aegypti</i> vs. <i>Anopheles freeborni</i> variando-se os parâmetros .....	44
Figura 29 – <i>F1-Score</i> do classificador <i>Aedes aegypti</i> vs. <i>Anopheles freeborni</i> variando-se os parâmetros .....	44
Figura 30 - Precisão do modelo multiclasse com conjunto de parâmetros 3 .....	46
Figura 31 - <i>Recall</i> do modelo multiclasse com conjunto de parâmetros 3.....	46
Figura 32 – <i>F1-score</i> do modelo multiclasse com conjunto de parâmetros 3.....	47

Figura 33 - Precisão do modelo multiclasse com conjunto de parâmetros 8 .....	47
Figura 34 - <i>Recall</i> do modelo multiclasse com conjunto de parâmetros 8.....	48
Figura 35 - <i>F1-score</i> do modelo multiclasse com conjunto de parâmetros 8.....	48
Figura 36 - Acurácia dos modelos multiclases .....	49
Figura 37 - Indicadores de desempenho classificador binário <i>Aedes aegypti</i> vs. demais espécies	50
Figura 38 - Erro no conjunto de treinamento e validação a cada iteração.....	50
Figura 39 - Acurácia no conjunto de treinamento e validação a cada iteração .....	51
Figura 40 - Acurácia dos classificadores binários .....	52
Figura 41 - Precisão classificadores binários .....	52
Figura 42 - <i>Recall</i> classificadores binários.....	53
Figura 43 - <i>F1-score</i> classificadores binários.....	53
Figura 44 - Três configurações de ensemble testadas .....	54
Figura 45 - Comparação entre <i>ensemble</i> e classificador binário .....	55

## LISTA DE TABELAS

Tabela 1 - Número de gravações por espécies .....	30
Tabela 2 - Duração por espécie antes e depois da edição dos arquivos .....	35
Tabela 3 – Relação dos modelos e seus conjuntos de parâmetros.....	42
Tabela 4 - Macro média dos indicadores de desempenho multiclasse .....	45
Tabela 5 - Indicadores de desempenho classificador binário <i>Aedes aegypti</i> vs. demais espécies	49



## LISTA DE SIGLAS

ML: *Machine Learning*

RNA: *Rede Neural Artificial*

CNN: *Convolutional Neural Network*

AUC: *Area Under the Curve*

FP: *Falso Positivo*

FN: *Falso Negativo*

VP: *Verdadeiro Positivo*

VN: *Verdadeiro Negativo*

# SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>11</b>
<b>2 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>13</b>
2.1 FREQUÊNCIA DO BATIMENTO DAS ASAS DOS MOSQUITOS .....	13
2.2 ESPECTOGRAMA EM ESCALA MEL .....	14
2.2.1 <i>Parâmetros da transformada de Fourier</i> .....	15
2.1 MACHINE LEARNING.....	15
2.2 APRENDIZADO SUPERVISIONADO.....	16
2.3 REDES NEURAIS ARTIFICIAIS.....	17
2.3.1 <i>Redes neurais convolucionais</i> .....	18
2.4 FUNÇÕES DE ATIVAÇÃO.....	20
2.4.1 <i>Sigmóide</i> .....	20
2.4.2 <i>Softmax</i> .....	21
2.4.3 <i>ReLU</i> .....	21
2.5 VALIDAÇÃO CRUZADA <i>K-FOLD</i> .....	22
2.6 INDICADORES DE DESEMPENHO .....	23
2.6.1 <i>Acurácia</i> .....	23
2.6.2 <i>Precisão</i> .....	24
2.6.3 <i>Recall</i> .....	24
2.6.4 <i>F1-score</i> .....	24
2.6.5 <i>AUC</i> .....	25
<b>3 REVISÃO DE LITERATURA.....</b>	<b>26</b>
3.1 MOSQUITO E SEU ÁUDIO .....	26
3.2 CLASSIFICAÇÃO DE ÁUDIO UTILIZANDO <i>MACHINE LEARNING</i> .....	27
<b>4 METODOLOGIA .....</b>	<b>29</b>
4.1 DATASET .....	29
4.1.1 <i>Análise no domínio do tempo</i> .....	32
4.1.2 <i>Análise no domínio da frequência</i> .....	32
4.3 EXTRAÇÃO DE <i>FEATURES</i> .....	36
4.4 MODELO DE MACHINE LEARNING.....	38
4.4.1 <i>Classificador multiclasse</i> .....	39
4.4.2 <i>Classificador binário</i> .....	39
4.4.3 <i>Ensemble de classificadores binários</i> .....	39
<b>5 RESULTADOS E DISCUSSÕES.....</b>	<b>40</b>
5.1 MODELO DAS REDES NEURAIS CONVOLUCIONAIS .....	40
5.2 DESEMPENHO CLASSIFICADOR MULTICLASSE .....	45
5.3 DESEMPENHO CLASSIFICADOR BINÁRIO .....	49
5.3 DESEMPENHO <i>ENSEMBLE</i> DE CLASSIFICADORES BINÁRIOS.....	51
<b>6 CONCLUSÕES .....</b>	<b>56</b>
<b>7 PROPOSTA PARA TRABALHOS FUTUROS .....</b>	<b>57</b>

<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>58</b>
<b>APÊNDICE A - FORMA DE ONDA DE TODAS AS ESPÉCIES .....</b>	<b>61</b>
<b>APÊNDICE B: ESPECTROGRAMA DE TODAS AS ESPÉCIES.....</b>	<b>64</b>
<b>APÊNDICE C: MATRIZ DE CONFUSÃO DO CLASSIFICADOR MULTICLASSE .....</b>	<b>67</b>

## 1 INTRODUÇÃO

Os vetores são organismos vivos que transmitem doenças infecciosas, sendo o mosquito o agente mais conhecido. Mais de um bilhão de pessoas são infectadas e dessas, mais de um milhão morrem devido a doenças transmitidas por vetores. Entre essas doenças pode-se citar a malária, dengue, doença de Chagas e febre amarela. Outra preocupação crescente é que esses mosquitos estão se tornando resistentes a inseticidas, onde a maioria das espécies já apresenta resistência a diversas classes de inseticidas. Em 2012 a dengue foi classificada como a doença viral com maior potencial epidêmico no mundo, tendo o seu principal vetor o *Aedes aegypti*. O mesmo também é responsável pela transmissão da febre amarela, chikungunya e zika vírus (WORLD HEALTH ORGANIZATION, 2014).

O controle desses vetores é importante, pois para doenças Chagas e dengue, é a única maneira de proteger a população. Dentre as técnicas utilizadas para minimizar o impacto desses vetores tem-se o esforço de mapeamento do mesmo, onde utilizam-se informações geográficas para identificar regiões que apresentem características ecológicas propícias à vida dos vetores. Mesmo já sendo estabelecido que o controle de vetores é efetivo no controle das doenças, o mesmo é negligenciado. Fatores que contribuem para essa negligência são a complexidade técnica, custo e logística necessária (TOWNSON *et al.*, 2005). A escassez de dados de mapeamento desses mosquitos parece ser devido à dificuldade de adquirir os mesmos. Em regiões com poucos recursos, o processo de identificação envolve a captura e identificação manual, portanto, necessita de recursos humano e de tempo. Para que se possa combater esses vetores de forma efetiva é necessário ter dados completos e atualizados sobre o mapeamento dos mesmos. Dessa forma, necessita-se de novos métodos de vigilância, que sejam de baixo custo, capazes de identificar populações de mosquitos (MUKUNDARAJAN *et al.*, 2017).

Muitos estudos argumentam que algoritmos de *Machine Learning* baseados em aprendizado supervisionado são a melhor escolha quando deseja-se realizar a classificação automática de espécies. A desvantagem desses métodos é que é necessário um alto número de dados para treinar o sistema além de exigir bastante processamento computacional (ACEVEDO *et al.*, 2009). Nessa forma de aprendizado fornece-se ao algoritmo um conjunto de dados de entrada juntamente com a rotulação dos mesmos e o algoritmo busca encontrar uma função que seja capaz de mapear o conjunto de dados de entrada em seus respectivos rótulos.

Com base neste cenário, o objetivo deste trabalho é avaliar a possibilidade de identificação de mosquitos, em especial o *Aedes aegypti*, através do som de seu voo utilizando algoritmos de *Machine Learning*. Obtendo-se resultados relevantes pode-se desenvolver dispositivos ou aplicativos celulares para o mapeamento temporal e geoespacial desses vetores em uma escala global, a partir da captura e processamento de áudio.

Como *feature* usou-se o espectrograma do áudio, de forma que para um mesmo arquivo de áudio são gerados  $n$  espectrogramas, sendo  $n$  dependente dos parâmetros da transformada de Fourier utilizados ao calcular o espectrograma. Portanto, após adquirir o espectrograma transforma-se o problema de classificação de áudio em um problema de classificação de imagens.

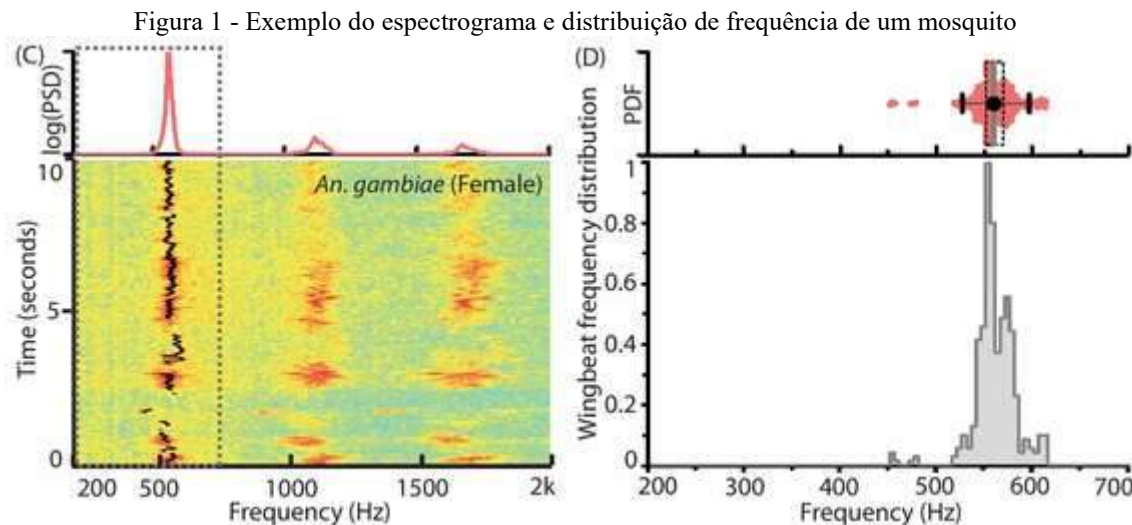
O trabalho está estruturado de forma que no Capítulo 2 tem-se um referencial teórico que aborda o som das asas de mosquitos e conceitos fundamentais de *Machine Learning*. No capítulo 3, apresenta-se uma relação de trabalhos relevantes envolvendo o estudo de sons de mosquitos e estudos onde se utilizou *Machine Learning* para realizar a classificação de gravações de áudio. No Capítulo 4 descreve-se a metodologia adotada seguido dos resultados que são expostos no Capítulo 5. Por fim, o Capítulo 6 apresenta uma conclusão baseada nos resultados obtidos.

## 2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta o embasamento teórico necessário para o trabalho.

### 2.1 FREQUÊNCIA DO BATIMENTO DAS ASAS DOS MOSQUITOS

Ao voar o mosquito produz um som composto primariamente por uma frequência fundamental. Esta frequência não é monótona e varia levemente ao longo do voo, tendo também harmônicas<sup>1</sup> de menor intensidade. A Figura 1 exibe o espectrograma de um mosquito fêmea da espécie *Anopheles gambiae* juntamente com a distribuição da frequência fundamental. A frequência de mosquitos fêmeas estão compreendidas em um intervalo de 200 a 700 Hz. Devido a maioria das espécies apresentar dimorfismo<sup>2</sup> sexual resulta-se em machos tendo uma frequência maior que fêmeas de mesma espécie (MUKUNDARAJAN *et al.*, 2017).



Fonte: Adaptado de MUKUNDARAJAN *et al.*, 2017.

De acordo com Jones (2012), a maioria dos livros classifica a frequência dos mosquitos como sendo de 400 a 600 Hz, devido a suas asas serem estreitas e pequenas. A frequência fica

<sup>1</sup> Dada uma frequência  $f$  a mesma possui harmônicas tais que  $1 \times f$  constitui a frequência fundamental,  $2 \times f$  a segunda harmônica,  $3 \times f$  a terceira harmônica, etc.

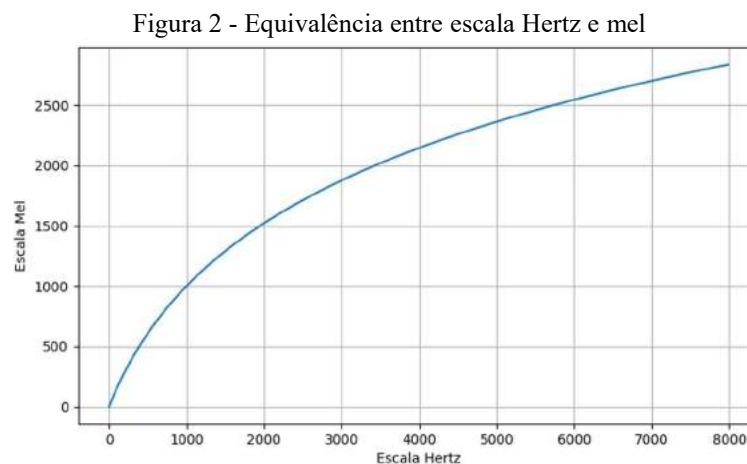
<sup>2</sup> Termo utilizado na biologia para categorizar quando em uma mesma espécie o macho e a fêmea apresentam características físicas, não sexuais, bastante distintas.

limitada à velocidade em que o sistema nervoso consegue gerar e disparar impulsos até as asas. Segundo White (1978 *apud* Leung *et al.*, 2000)<sup>3</sup> essa frequência varia de 250 a 600 Hz.

## 2.2 ESPECTOGRAMA EM ESCALA MEL

Espectrograma é uma representação visual do espectro das frequências ao longo do tempo, comumente gerado utilizando a transformada de Fourier. Em problemas de classificação de áudio usando *Machine Learning*, a maioria das pesquisas não usa o espectrograma em escala linear e sim utilizam a escala logarítmica ou a escala mel, pois essas reduzem a resolução das frequências altas se assemelhando com a percepção humana (DIELEMAN; SCHRAUWEN, 2014). Em 1937, Stevens, Volkman e Newman estabeleceram em seu estudo a escala mel, o nome foi derivado da palavra *melody*, que é baseada na percepção humana de frequência. Nesse estudo primeiro mapeou-se a frequência de 1.000 Hz como sendo 1000 unidades subjetivas (mel), então pediu-se a observadores que identificassem a frequência que parecesse ser metade da frequência inicial, sendo esse processo repetido para diversas frequências. Existem diversas interpretações de qual seria a escala mel. O'Shaughnessy (2000) define a conversão de Hertz para mel pela Equação 1 e a mesma é representada visualmente pela Figura 2.

$$mel = 2595 \log_{10} \left( 1 + f/700 \right) \quad (1)$$

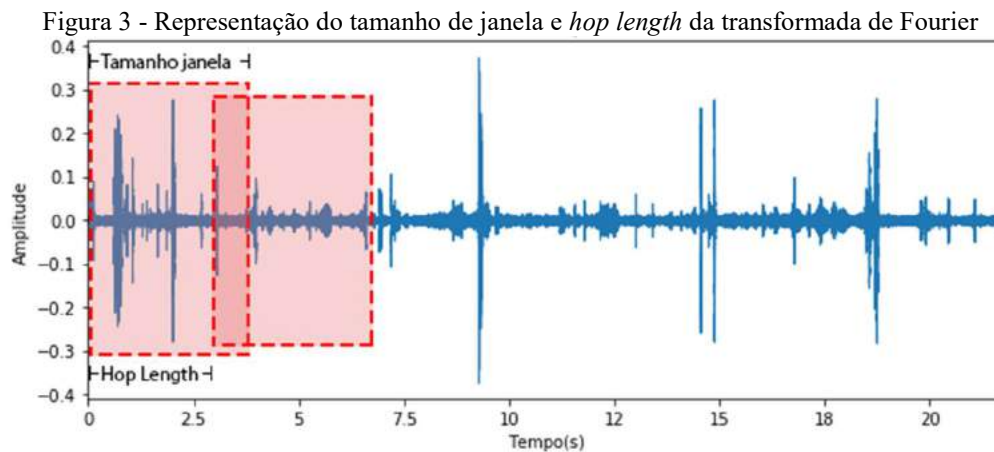


Fonte: O autor, 2018.

<sup>3</sup> WHITE, William. **A mosquito is Born**. New York: Sterling, 1978. p. 21. *apud* LEUNG *et al.*

### 2.2.1 Parâmetros da transformada de Fourier

Ao se obter o espectrograma por meio da transformada de Fourier é necessário especificar três parâmetros: tamanho da janela, *hop length* e número de bandas. O tamanho da janela é o trecho do sinal que será analisado. O parâmetro *Hop length* é a distância entre uma janela e outra. Sendo o último parâmetro o número de bandas, que define a resolução no domínio da frequência. O conceito de tamanho de janela e *hop length* (também chamado de *hop size*) são exemplificados na Figura 3 (SMITH III; SERRA, 1987).



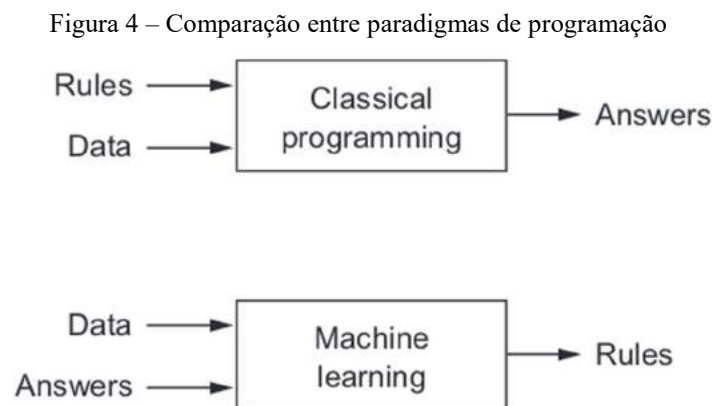
Fonte: O autor, 2018.

## 2.1 MACHINE LEARNING

O termo *Machine Learning* (ML), em português Aprendizado de Máquina, é utilizado quando se refere a programas de computador que são capazes aprender a realizar alguma tarefa com base na experiência. Mitchell (1997, p. 2) define essa capacidade de aprender como “Diz-se que um programa de computador aprende com experiência E em relação a alguma classe de tarefas T e medida de desempenho P, se seu desempenho nas tarefas T, conforme medido por P, melhora com a experiência E”. Tendo-se uma quantidade suficiente de dados representando o domínio de interesse, é possível rotular a saída esperada, constituindo um aprendizado supervisionado, e fazer com que o computador identifique um padrão e automaticamente extraia um algoritmo para solucionar essa tarefa. Esse algoritmo é constituído de um modelo, definido por determinados parâmetros. A tarefa de treinamento desse modelo consiste em otimizar os parâmetros do mesmo



usando os dados que constituem o problema (ALPAYDIN, 2014). Assim, um sistema que utiliza *Machine Learning* é treinado ao invés de ser explicitamente programado: o sistema recebe múltiplos exemplos relevantes a uma tarefa e acha uma estrutura estatística nesses exemplos que permite ao mesmo criar regras para automatizar a tarefa (CHOLLET, 2017). A Figura 4 representa uma comparação entre o paradigma de programação clássico com o utilizado em *Machine Learning*.



Fonte: CHOLLET, 2017.

## 2.2 APRENDIZADO SUPERVISIONADO

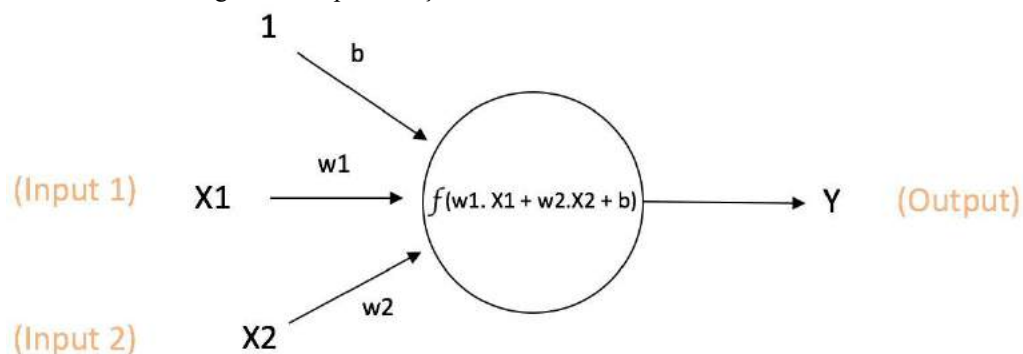
Uma das subclassificações de ML é baseada no tipo de aprendizado utilizado, o qual pode ser classificado entre dois paradigmas principais: Supervisionado e Não Supervisionado. No aprendizado supervisionado, no qual se baseia o presente trabalho, o sistema tem uma entrada e deve classificar a mesma utilizando rótulos, que também são referidos como classes. É tarefa do supervisor do algoritmo separar o *dataset* fornecido em conjuntos chamados de dados de treinamento e dados de validação. Ao realizar o treinamento deve-se prover os dados juntamente com os rótulos dos mesmos, de forma que o sistema aprenda padrões que posteriormente serão utilizados para classificar dados não vistos durante a etapa de treinamento (SHALEV-SCHWARTZ; BEN-DAVID, 2014). Caso o modelo apresente um ótimo desempenho na etapa de treinamento, mas ao ser alimentado com dados nunca vistos o desempenho não seja tão alto, tem-se o caso de *overfitting*. Isso ocorre porque o modelo começa a utilizar ruído presente nos dados de treinamento como característica para realizar a classificação, porém esse ruído não está presente em novos dados, portanto prejudicando o desempenho do mesmo (CHOLLET, 2017). Dessa forma,

o modelo ideal deve ser genérico o suficiente para que o mesmo apresente métricas de desempenho semelhantes na etapa de treinamento e ao ser exposto a novos dados.

### 2.3 REDES NEURAIS ARTIFICIAIS

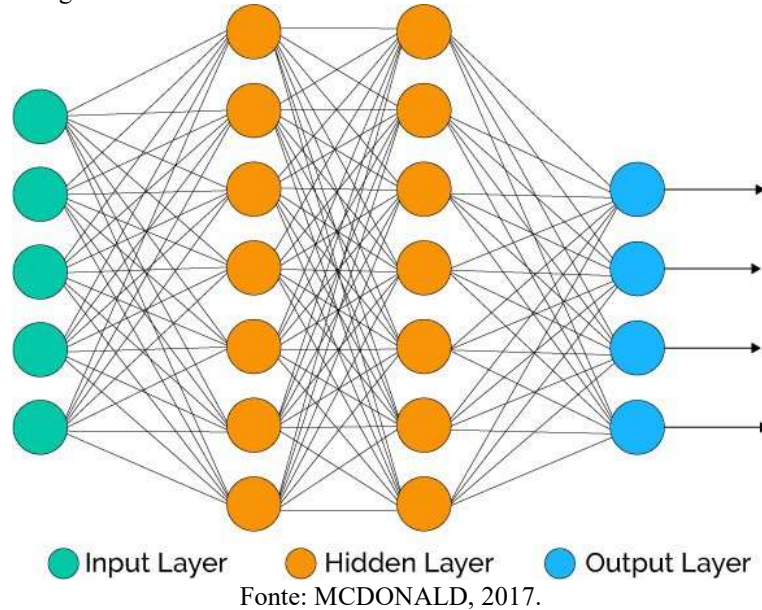
Redes neurais artificiais (RNAs) são um dos métodos mais utilizados para resolver problemas que envolvem interpretar dados complexos de sensores do mundo real. Esse modelo inspira-se no fato de que sistemas de aprendizado biológicos são compostos de redes complexas de neurônios. Redes neurais são compostas por um conjunto de unidades que são interconectadas entre si, onde cada unidade recebe como entrada um conjunto de valores reais (possivelmente a saída de outras unidades) e produz como saída apenas um valor real (que pode ser usado como entrada para diversas outras unidades). Essa unidade, representada na Figura 5 e chamada de neurônio, realiza uma soma ponderada dos valores em sua entrada somando-se também uma constante chamada de *bias*. O resultado é utilizado como entrada para uma função de ativação, que irá retornar o valor de saída do neurônio (MITCHELL, 1997). A rede neural é dividida em camadas, devendo obrigatoriamente conter uma camada de entrada e uma de saída, e podendo conter  $n$  camadas intermediárias que são chamadas de camadas ocultas. Cada neurônio da camada posterior está completamente conectado aos neurônios da camada anterior. A Figura 6 exibe uma rede neural cuja camada de entrada contém 5 neurônios, seguido por duas camadas ocultas com 7 neurônios cada, e uma camada de saída realiza a classificação de 4 classes.

Figura 5 - Representação de um neurônio de uma rede neural



Fonte: KARN, 2016.

Figura 6 - Rede neural multiclasse contendo duas camadas ocultas



### 2.3.1 Redes neurais convolucionais

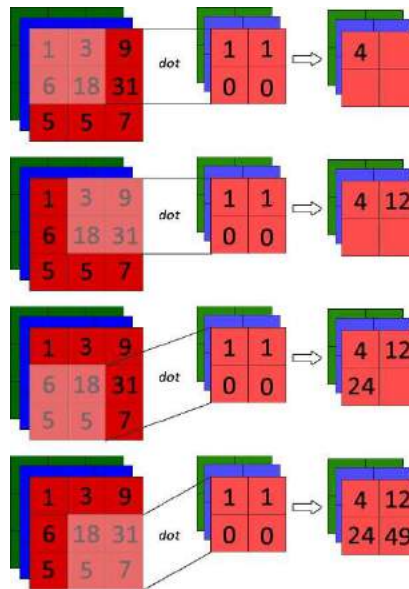
Redes neurais convolucionais (do inglês *Convolutional Neural Network*, CNN), são modelos quase que universalmente usados em aplicações que envolvem visão computacional (CHOLLET, 2017). Diferente de redes neurais ordinárias, as redes neurais convolucionais explicitamente assumem em sua arquitetura que a entrada será tridimensional, permitindo que certas propriedades sejam implementadas em sua arquitetura. Nessa arquitetura, cada camada tem seus neurônios organizados em 3 dimensões: largura, altura e profundidade. Os neurônios de uma camada também se conectam a apenas uma região da camada anterior, ao invés de serem totalmente conectados como é o caso das redes neurais ordinárias. A última camada será um vetor no formato  $1 \times 1 \times N$ , onde  $N$  é o número de classes do classificador (LI; JOHNSON; YEUNG, 2018a). Diferente de uma RNA, cada camada de uma CNN tem um comportamento diferente, sendo as mais utilizadas:

- **Camada convolutiva:** essa camada é a mais importante da rede, é composta por um conjunto de filtros (*kernels*). Os filtros são matrizes pequenas (por exemplo,  $5 \times 5 \times 3$ ) que se movem ao longo da largura e altura da imagem, realizando a operação de convolução por meio do produto escalar dos pixels que estão englobados pelo mesmo, obtendo assim um mapa de características bidimensional que é a resposta do filtro sobre cada região da imagem. Os valores desses filtros não são definidos, eles se alteram ao longo da etapa de

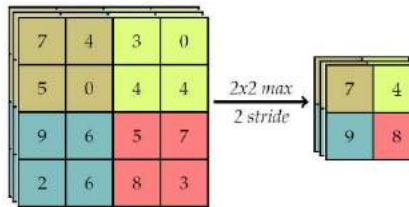
treinamento buscando o valor com a menor taxa de erro. O deslizamento desse filtro pode ser controlado por um parâmetro chamado de *stride*: quando o mesmo é 1 significa que o filtro está sendo movido um pixel por vez. Também se tem como parâmetro o número de filtros que se deseja extrair, onde cada filtro corresponde a uma característica que a rede irá aprender sobre a imagem. Portanto, é importante que os valores dos filtros sejam inicializados utilizando diferentes critérios, porque caso sejam inicializados de forma similar pode ocorrer o caso em que dois filtros aprendam a mesma característica (PACHECO, 2017). A Figura 7 exemplifica a operação dessa camada em uma imagem  $3 \times 3 \times 3$  utilizando um filtro  $2 \times 2 \times 3$  e *stride* 1. Ao percorrer a imagem utilizando filtros, a rede aprende padrões locais equivalentes ao tamanho do filtro. Uma vez que esses padrões tenham sido aprendidos pode-se identificar os mesmos em qualquer lugar da imagem. Uma primeira camada convolutiva aprende alguns padrões locais como arestas, camadas convolutivas subsequentes irão utilizar os padrões detectados pelas camadas anteriores para aprender novas características ainda mais complexas, dessa forma permitindo que uma CNN seja capaz de identificar conceitos visuais complexos e abstratos (CHOLLET, 2017).

- **Camada de *pooling*:** tem como objetivo reduzir o tamanho das matrizes resultantes da convolução, reduzindo a quantidade de parâmetros aprendidos pela rede e permitindo um controle do *overfitting*. Seus parâmetros são a largura, altura e *stride* do filtro. A profundidade mantém-se inalterada. A saída dessa camada será o maior valor englobado pela matriz do filtro. Usualmente se utiliza um filtro  $2 \times 2$  com *stride* 2, diminuindo o tamanho da camada anterior em 75% conforme a Figura 8 (LI; JOHNSON; YEUNG, 2018b).
- ***Dropout*:** durante o treinamento, aleatoriamente elimina-se (configurando os valores para zero) alguns neurônios da camada anterior. Possui um único parâmetro que é a probabilidade de um neurônio ser eliminado, chamado de taxa de *dropout*, usualmente definida entre os valores 0,2 e 0,5. Pode parecer contra intuitivo, porém adicionando essa camada de *dropout* ajuda a introduzir ruído no modelo, que por sua vez ajuda a quebrar padrões que não sejam significativos, de forma que não ocorra *overfitting* (CHOLLET, 2017).

Figura 7 - Exemplo da operação de convolução



Fonte: PACHECO, 2017.

Figura 8 - Operação de *pooling* em uma imagem 4×4 usando um filtro 2×2 com *stride* 2

Fonte: PACHECO, 2017.

## 2.4 FUNÇÕES DE ATIVAÇÃO

Funções de ativação são utilizadas pelos neurônios após realizar a soma ponderada de suas entradas. Uma das motivações é adicionar uma operação não linear, caso contrário, por mais complexo que fosse a rede construída, ela ainda seria apenas uma série de transformações lineares (CHOLLET, 2017).

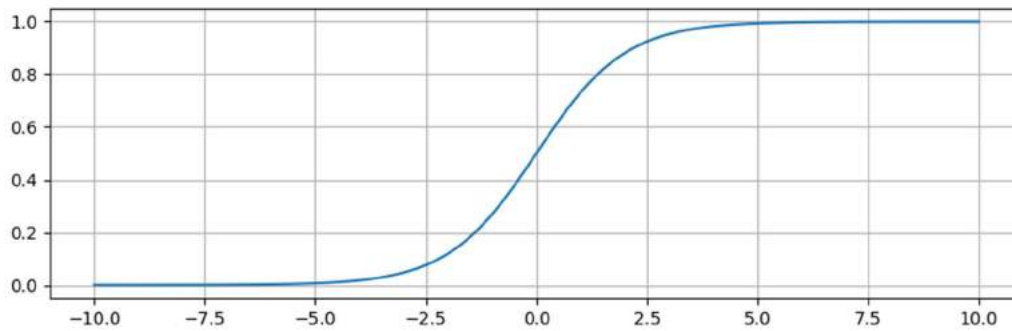
### 2.4.1 Sigmóide

Essa função dada pela Equação 2 e representada pela Figura 9, apresenta uma saída que varia de 0 a 1 (LI; JOHNSON; YEUNG, 2018a). No caso de classificadores é útil para a última

camada da rede onde se deseja a probabilidade de a entrada permanecer a cada classe, onde valores próximos de 0 significam que a probabilidade é baixa e valores próximos de 1 uma alta probabilidade.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

Figura 9 - Função de ativação sigmóide



Fonte: O autor, 2018.

#### 2.4.2 Softmax

Utilizado em classificadores multiclasse, onde sua saída é uma distribuição probabilística onde cada classe varia de 0 a 1 e a soma de todas as classes obrigatoriamente será 1. Calculada utilizando a Equação 3,

$$\sigma(x)_j = \frac{e^{x_j}}{\sum_{k=1}^K e^{x_k}} \text{ para } j = 1, \dots, K \quad (3)$$

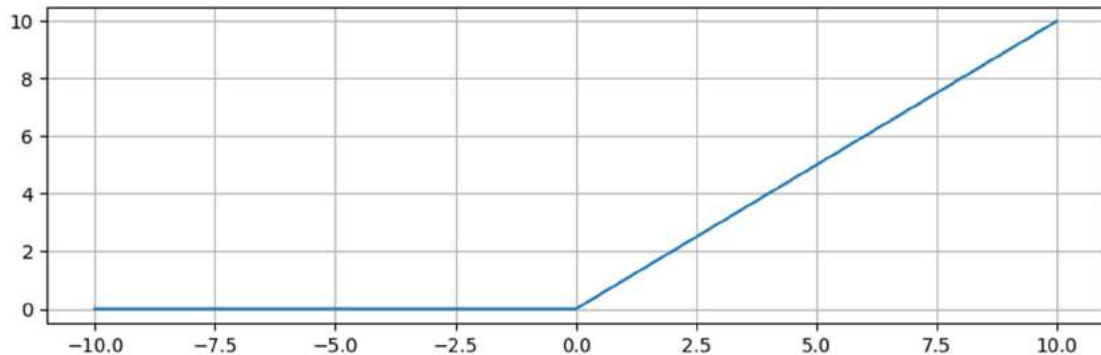
onde K é o comprimento do vetor  $x$  (LAN, 2017).

#### 2.4.3 ReLU

Chamada de unidade linear retificada, é definida pela Equação 4 e seu gráfico demonstrado na Figura 10. A função apenas limita valores menores que zero para zero. Tem se tornado muito popular nos últimos anos, tendo como uma de suas vantagens o custo computacional baixo em relação à função sigmóide (LI; JOHNSON; YEUNG, 2018a).

$$f(x) = \max(0, x) \quad (4)$$

Figura 10 - Função de ativação ReLU



Fonte: O autor, 2018.

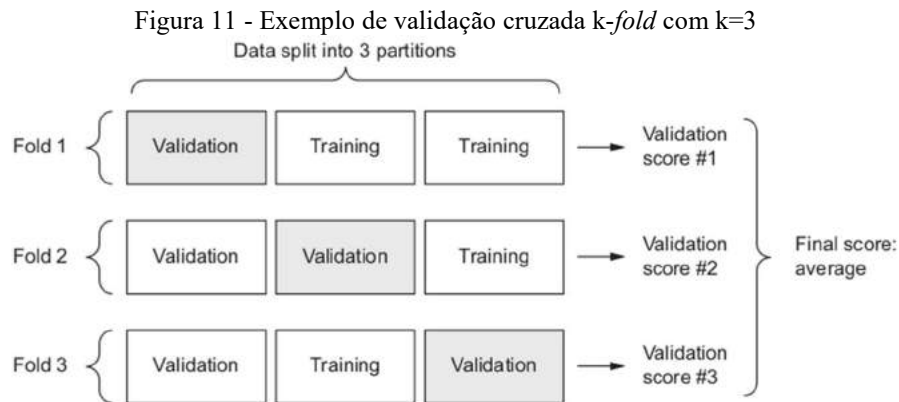
## 2.5 VALIDAÇÃO CRUZADA *K-FOLD*

Usualmente, para avaliar o desempenho de um classificador divide-se o *dataset* em três conjuntos: treinamento, validação e teste, onde o conjunto de treinamento é utilizado para treinar o modelo, validação para otimizar o mesmo (ajuste de parâmetros) e o de teste deve ser usado apenas após o modelo atingir um resultado satisfatório no conjunto de validação. Se utiliza dessa estratégia para que o modelo não faça *overfitting* sobre os dados. Deve-se tomar cuidado, pois cada vez que se ajusta o modelo baseado no resultado do conjunto de validação na realidade se está suprimindo indiretamente informações sobre o mesmo, por isso utiliza-se um terceiro conjunto de teste que deve ser utilizado o mínimo possível (CHOLLET, 2017).

Nos casos em que não se possui muitos dados o conjunto de validação será pequeno, de forma que os indicadores de desempenho podem variar muito dependendo dos dados que forem utilizados para treinamento e validação, sendo difícil avaliar o modelo de maneira confiável. Para contornar esse problema pode-se utilizar de uma técnica conhecida como validação cruzada *k-fold*, onde se divide o conjunto de dados original em K subconjuntos de tamanho semelhantes. Então realiza-se o treinamento e validação K vezes, sendo que a cada iteração será utilizado um subconjunto diferente para a validação e os demais subconjuntos serão usados para treinamento. Os indicadores de desempenho do modelo serão as médias dos indicadores de desempenho obtidos em cada iteração (CHOLLET, 2017).

Na prática, é comum a utilização de 10-*folds* (k=10), de forma que 10% dos dados serão utilizados para validação. Em certos cenários há o desbalanço entre as classes, nessas situações é

recomendado utilizar uma validação cruzada *k-fold* estratificada. Essa validação divide os subconjuntos de forma que cada subconjunto mantenha a proporção para cada classe em relação ao conjunto de dados original (REFAEILZADEH; TANG; LIU, 2009).



Fonte: CHOLLET, 2017.

## 2.6 INDICADORES DE DESEMPENHO

Dado um classificador e um conjunto de instâncias a serem analisadas, usualmente como dados de validação ou de teste, pode-se construir uma matriz de confusão. Essa matriz é a representação base da qual são extraídas métricas de desempenho como acurácia, precisão, *recall* e *F1-score* (FAWCETT, 2005). A Figura 12 apresenta uma matriz de confusão binária.

Figura 12 - Matriz de confusão de um classificador binário

	Preveu-se “Classe Negativa”	Preveu-se “Classe Positiva”
Classe Negativa	Verdadeiro Negativo (VN)	Falso Positivo (FP)
Classe Positiva	Falso Negativo (FN)	Verdadeiro Positivo (VP)

Fonte: Adaptado de GEITGEY, 2016.

### 2.6.1 Acurácia

A acurácia é o indicador do número de instâncias que a rede previu o valor corretamente e é calculado utilizando a Equação 5. Em algumas aplicações o número de exemplos em cada classe é desbalanceado e nesses cenários uma acurácia alta pode ser um indicador ilusório do



desempenho. Isso fica evidente quando se imagina um cenário onde 2% das instâncias são positivas e 98% são negativas, se um classificador classificar todas as entradas como sendo negativas o mesmo terá 98% de acurácia, porém sem nunca classificar uma instância positiva ele é um classificador inútil (KUBAT, 2015).

$$A = \frac{VP + VN}{VP + FP + VN + FN} \quad (5)$$

### 2.6.2 Precisão

Dada pela Equação 6, calcula a porcentagem de verdadeiros positivos em relação a todas as entradas que o classificador classificou como sendo positivo. Em outras palavras, precisão é a probabilidade do classificador estar correto ao identificar algo como sendo positivo (KUBAT, 2015).

$$P = \frac{VP}{VP + FP} \quad (6)$$

### 2.6.3 Recall

Esse indicador é a porcentagem de instâncias positivas corretamente identificadas pelo classificador. Portanto, é calculado tomando-se o número de verdadeiros positivos e os dividindo pelo número total de amostras positivas presentes no *dataset* conforme a Equação 7 (KUBAT, 2015).

$$R = \frac{VP}{VP + FN} \quad (7)$$

### 2.6.4 F1-score

Devido ao fato da precisão e *recall* serem indicadores de desempenho diferentes e algumas vezes contraditórios, surgiram algumas tentativas de agrupar os dois indicadores em um único indicador de desempenho. A fórmula mais utilizada,  $F_\beta$  é dada pela Equação 8,

$$F_{\beta} = \frac{(\beta^2 + 1) \times \text{precisão} \times \text{recall}}{(\beta^2 \times \text{precisão}) + \text{recall}} \quad (8)$$

onde  $\beta \in [0, \infty)$  e serve como um peso que determina a importância de cada critério. Um valor maior que 1 irá dar mais peso ao *recall* enquanto que um valor menor que 1 irá favorecer a precisão. Em seus extremos 0 e  $\infty$  esse indicador converge para precisão e *recall* respectivamente. Na maioria dos casos não se sabe como priorizar entre os dois parâmetros, portanto é comum usar o parâmetro  $\beta = 1$  (KUBAT, 2015).

### 2.6.5 AUC

*Area Under the Curve* (AUC) geralmente refere-se a área da curva ROC (*Receiver Operating Characteristic*). O eixo Y dessa curva é composto pelo *recall* enquanto o eixo X é dado pela Equação 9 (D'SOUZA, 2018).

$$\text{Taxa de falso positivo} = \frac{FP}{VN + FP} \quad (9)$$

Então calcula-se o *recall* e a taxa de falso positivo variando-se o *threshold* de 0 a 1. Onde um *threshold* “0” considera todas as entradas como sendo instâncias positivas, enquanto um *threshold* “1” considera todas como sendo negativas. Então plota-se a curva, a área abaixo da mesma é uma representação do desempenho do classificador, seu valor podendo variar entre 0 e 1. Um modelo que erra todas as previsões terá AUC 0, enquanto um modelo que acerta todas as previsões terá AUC 1 (D'SOUZA, 2018).

### 3 REVISÃO DE LITERATURA

Este capítulo apresenta trabalhos relacionados com o tema abordado.

#### 3.1 MOSQUITO E SEU ÁUDIO

Em 1945, Kahn, Celestin e Offenhauser buscaram entender a forma em que os mosquitos comunicam o acasalamento para então reproduzir esse som e atrair os insetos para alguma armadilha e matá-los. Identificou-se que os tons fundamentais variam de 250 a 1500 Hz, porém isso abrange o barulho dos mosquitos enquanto estão voando, o bater das asas ao estarem em repouso, o esfregar das asas e um som cuja origem não foram capazes de identificar. Offenhauser e Kahn (1949) realizaram mais gravações, inclusive comparando o som de dois *Aedes aegypti* (fêmea) que foram criados ao mesmo tempo em ambientes idênticos, sendo a única diferença significativa que um dos espécimes foi exposto à fósforo radiativo por cerca de 3 dias enquanto ainda era uma larva. Notou-se claramente diferença entre os dois insetos, e Offenhauser e Kahn concluíram que para ser capaz de diferenciar espécies seria necessário um grande número de gravações, sobre uma grande variedade de condições controladas e repetidas frequentemente para que se tenha relevância estatística.

Mukundarajan *et al.* (2017) verificaram que é possível realizar gravações do voo de mosquitos com alta qualidade utilizando telefones celulares ordinários, mesmo havendo ruídos de fundo. Constatou-se que o efeito Doppler é desprezível visto que mosquitos dificilmente voam a mais de um metro por segundo, sendo então a diferença da frequência observada para a da fonte de menos que 2 Hz, e em um único voo o mosquito pode mudar sua frequência fundamental em até 100Hz. As gravações de celulares se beneficiam do fato de que os microfones dos mesmos são projetados de forma a ter sensibilidade máxima na faixa de frequência da voz humana, compreendida entre 300 a 3000Hz, portanto coincidindo com a frequência de voo dos mosquitos. A utilização de celulares também possibilita a aquisição de metadados de localização e horário em que a gravação foi realizada. Essas informações auxiliam na correta identificação da espécie, pois algumas espécies têm hábitos noturnos e outras possuem hábitos diurnos, além de certas espécies estarem contidas em apenas uma região do globo. Adicionalmente o uso destes dados permite a realização do mapeamento espacial e temporal por espécie. Neste trabalho realizaram-se as

gravações de três modos diferentes: mosquito contido em caixas próprias para o mesmo, voo livre e em recipientes fechados ordinários (copos, garrafas e sacos plásticos). Provou-se que aparelhos celulares são capazes de gravar o bater de asas de mosquito, com qualidade que ainda seja possível identificar o bater das mesmas, se o aparelho estiver até a 50 mm de distância do mosquito, sendo essa distância dobrada em aparelhos mais modernos. Gravações com diferentes aparelhos celulares apresentam espectro acústico similar, permitindo a identificação sem que a mesma seja dependente do modelo utilizado. Como resultado, identificou-se que estirpes<sup>4</sup> apresentam a mesma frequência fundamental para o *Anopheles gambiae* e *Aedes aegypti*, porém isso não foi observado no *Anopheles arabiensis* e destacou-se que esse caso deve ser investigado mais a fundo.

### 3.2 CLASSIFICAÇÃO DE ÁUDIO UTILIZANDO *MACHINE LEARNING*

Cakir *et al.* (2017) buscaram desenvolver um modelo que classifica se um segmento de áudio pertence ou não a um pássaro, as mesmas técnicas podem ser aplicadas para realizar a classificação de mosquitos. Conforme destacado no trabalho, isso poderia servir como uma primeira camada para um segundo modelo que identificaria a espécie do pássaro. Sons de pássaros podem ser separados entre vocais e não vocais (bater de um pica-pau), e devido à dificuldade de se classificar os sons não vocais focou-se somente nos sons vocais. A primeira etapa consistiu em extrair como *feature* o espectrograma em escala mel utilizando a biblioteca librosa. Foram utilizadas 40 bandas, 40 ms de áudio e 50% de sobreposição entre uma amostra e outra. Os sons de pássaros estão contidos em uma faixa de frequência de 2 – 8 kHz, porém obteve-se resultados superiores quando se utilizou todo o espectro de 0 Hz à 22 kHz. A arquitetura da rede consiste em uma CNN onde se tem uma sequência de camada convolucional e *max pooling* que é repetida quatro vezes, então se faz uma pilha de todas as amostras de espectro obtidas para um determinado arquivo e usa-se essa pilha como entrada para outras redes cuja arquitetura é baseada em uma rede neural. No fim o modelo conseguiu AUC *score* de 88,5%.

Dieleman e Schrauwen (2014) utilizaram o *dataset* Magnatagatune que contém 25.863 arquivos de música, cada qual com 29 segundos separados em 188 categorias. O objetivo é rotular automaticamente sons musicais nas mais diversas categorias (clássica, vocal masculino, jazz, etc.).

---

<sup>4</sup> Grupo de descendentes que compartilham um ancestral comum que compartilham semelhanças morfológicas ou fisiológicas. No caso do trabalho são mosquitos que desenvolveram uma mutação a qual são resistentes a alguns inseticidas.

Usou-se apenas as 50 categorias mais populares de acordo com os dados contidos no *dataset* e comparou-se o desempenho de duas CNNs, a primeira usando como *feature* o espectrograma e a segunda utilizando a forma de onda. A primeira obteve uma média AUC de 0,876 e já a segunda rede obteve 0,840, portanto ambas *features* apresentam bons resultados, sendo o espectrograma superior.

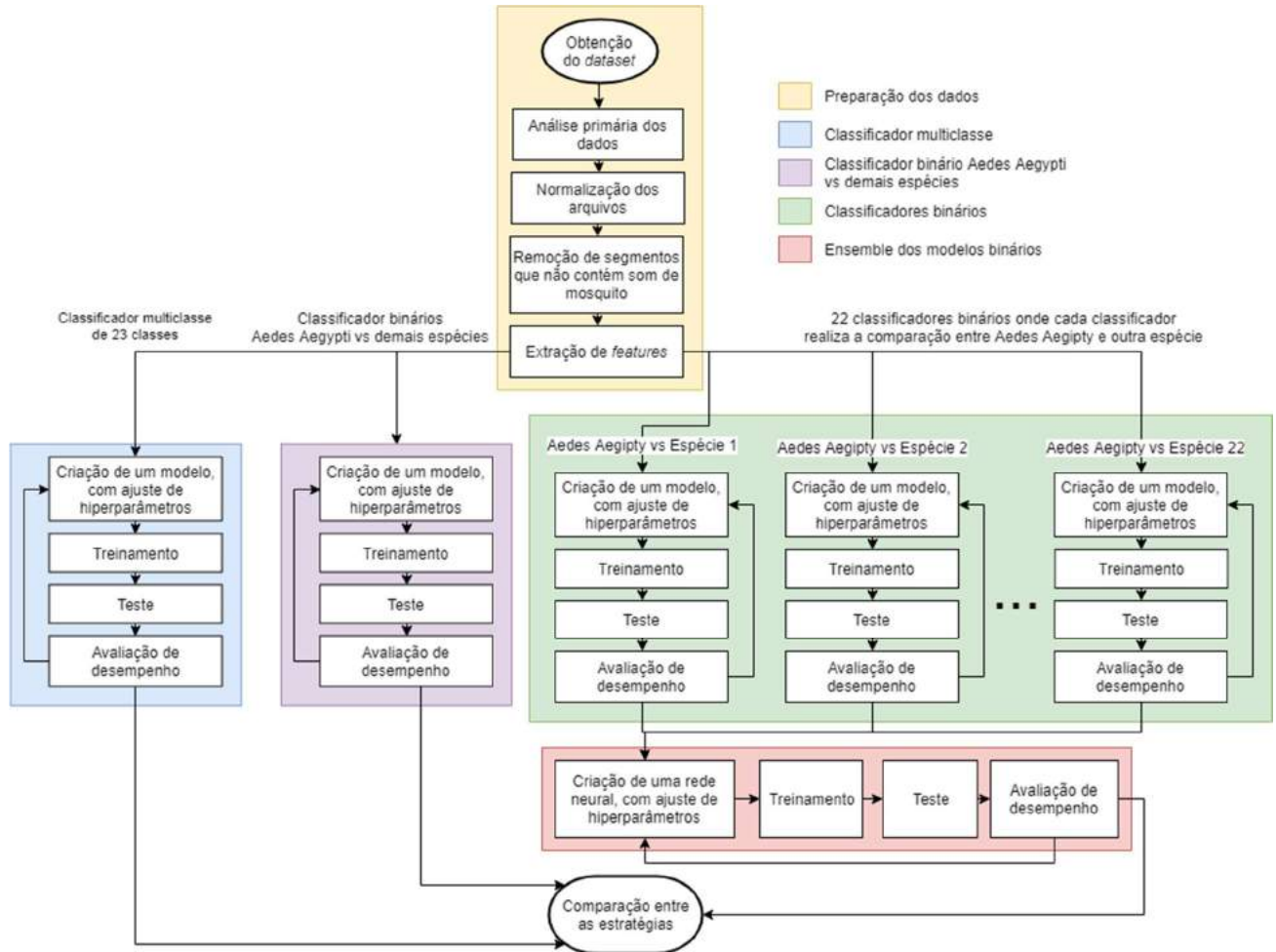
Outro exemplo de utilização de CNNs para classificação de áudios é a pesquisa realizada por Piczak (2015), que aplicou esse modelo em três diferentes *datasets* de sons urbanos. Tendo como objetivo classificar um segmento de áudio em diferentes categorias (buzina de carro, cachorro latindo, sirenes, etc.). Todas as gravações foram reamostradas em 22.050 Hz, sendo utilizado o espectrograma em escala mel com tamanho de janela igual a 1024, *hop length* de 512 e 60 bandas. Tentou-se duas abordagens, a primeira com 41 quadros e 50% de sobreposição e a segunda com 101 quadros e 90% de sobreposição. Além do espectrograma também se usou como *feature* o delta do espectrograma, que é a derivada do mesmo. A arquitetura da rede consiste em uma camada convolutiva de 80 filtros retangulares (57 por 6 com *stride* 1×1), seguido de *dropout* e *maxpooling*, uma segunda camada convolutiva cujo filtro tem tamanho 1×3 e depois duas camadas totalmente conectadas com *dropout*. O desempenho equivale-se ao obtido quando se utilizou *features* cuidadosamente selecionadas por especialistas. Como exemplo tem-se a acurácia em um dos *datasets* que se obteve 73,1% sendo o melhor desempenho obtido até então de 73,7%.

De acordo com Rothmann (2018), não é correto utilizar CNNs e o espectrograma para classificar áudios visto que em uma imagem o valor de um pixel geralmente corresponde a cor de um único objeto, já em um espectrograma a magnitude de um dado pixel pode ter sido produzida por frequências sobrepostas. Os eixos do espectrograma também contêm informações diferentes, sendo um a intensidade da frequência e o outro o tempo. Por CNNs aprenderem padrões locais, elas são capazes de identificar um padrão independentemente da posição do mesmo na imagem, porém em um espectrograma, um mesmo padrão algumas frequências acima pode ser a diferença entre a voz de um homem e uma criança. Essa argumentação é válida, porém os bons resultados das pesquisas que usam CNN são suficientes para provar que por enquanto essa é uma das melhores, se não a melhor, abordagem a ser seguida.

## 4 METODOLOGIA

Neste capítulo apresenta-se a metodologia utilizada no trabalho. O fluxograma geral pode ser visualizado na Figura 13.

Figura 13 - Fluxograma da metodologia



Fonte: O autor, 2018.

### 4.1 DATASET

O *dataset* contendo 1.285 arquivos de áudio e categorizados em 20 espécies foi obtido graças ao projeto Abuzz de Mukundarajan *et al.* (2017), realizado pelo laboratório Prakash de Stanford, que realizou gravações com diferentes aparelhos celulares e as disponibilizou gratuitamente. As espécies *Anopheles arabiensis* e *Anopheles gambiae* também são subclassificadas em diferentes estirpes. Esse tipo de subclassificação é interessante para verificar

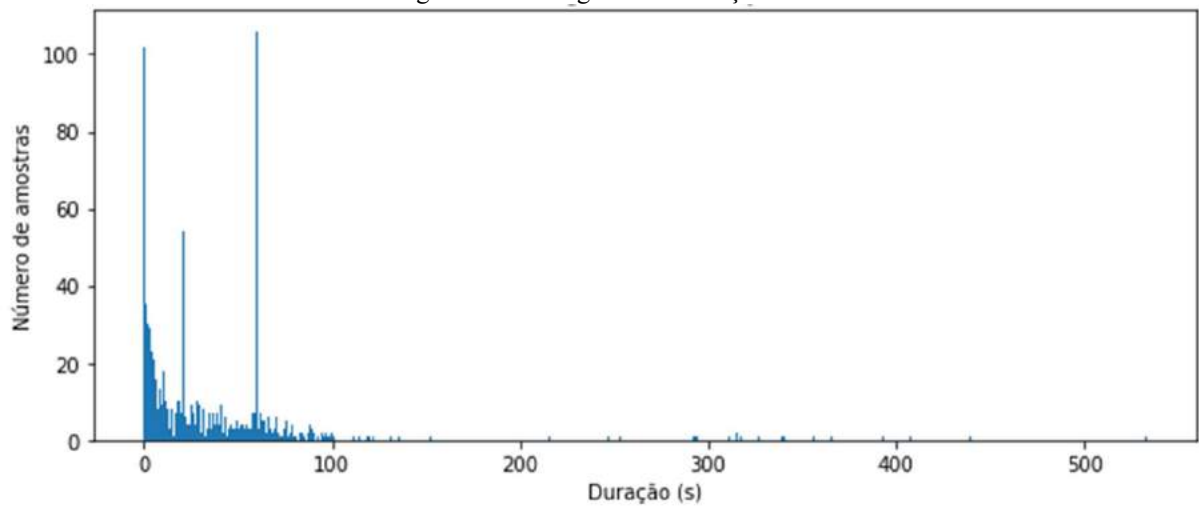
se as mesmas apresentam características sonoras distintas. A Tabela 1 mostra as espécies e estirpes presentes no *dataset* juntamente com o número de gravações e a duração total das mesmas.

Tabela 1 - Número de gravações por espécies

Espécie	Número de arquivos	Duração total dos áudios (segundos)
<i>Aedes aegypti</i>	22	1.736,87
<i>Aedes albopictus</i>	7	966,37
<i>Aedes mediovittatus</i>	3	180,00
<i>Aedes sierrensis</i>	361	3.337,58
<i>Anopheles albimanus</i>	40	1.736,55
<i>Anopheles arabiensis dongola</i>	6	1.040,11
<i>Anopheles arabiensis rufisque</i>	7	943,76
<i>Anopheles atroparvus</i>	7	907,64
<i>Anopheles dirus</i>	65	1.733,48
<i>Anopheles farauti</i>	47	1.727,52
<i>Anopheles freeborni</i>	54	2.397,82
<i>Anopheles gambiae akron</i>	7	972,98
<i>Anopheles gambiae kisumu</i>	57	1.747,59
<i>Anopheles gambiae rsp</i>	2	567,66
<i>Anopheles merus</i>	5	300,00
<i>Anopheles minimus</i>	68	2.923,11
<i>Anopheles quadriannulatus</i>	7	1.0178,10
<i>Anopheles quadrimaculatus</i>	6	665,71
<i>Anopheles stephensi</i>	58	2.909,69
<i>Culex pipiens</i>	9	710,37
<i>Culex quinquefasciatus</i>	13	758,13
<i>Culex tarsalis</i>	12	613,87
<i>Culiseta incidens</i>	37	2.971,58

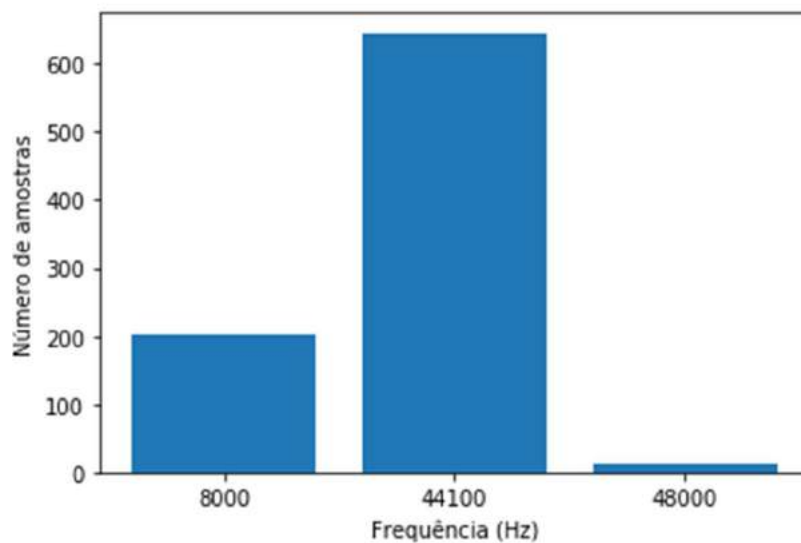
A análise dos dados foi realizada em Python utilizando principalmente as bibliotecas librosa (análise de áudio) (MCFELL, 2015), numpy (computação científica) e matplotlib (geração de gráficos). Os arquivos foram gravados utilizando diferentes aparelhos de celular em diferentes formatos (WAV, MP4, M4A e AMR), sendo que a maioria das espécies apresenta gravações com dois ou mais celulares. A duração dos mesmos e a frequência de amostragem variam, conforme pode ser visto nas Figuras 14 e 15 respectivamente.

Figura 14 - Histograma da duração



Fonte: O autor, 2018.

Figura 15 - Histograma da frequência de amostragem



Fonte: O autor, 2018.

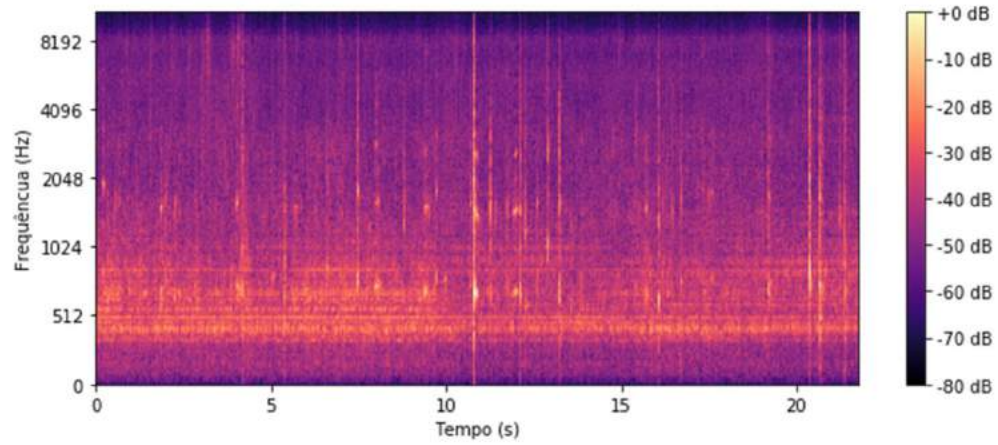


#### 4.1.1 Análise no domínio do tempo

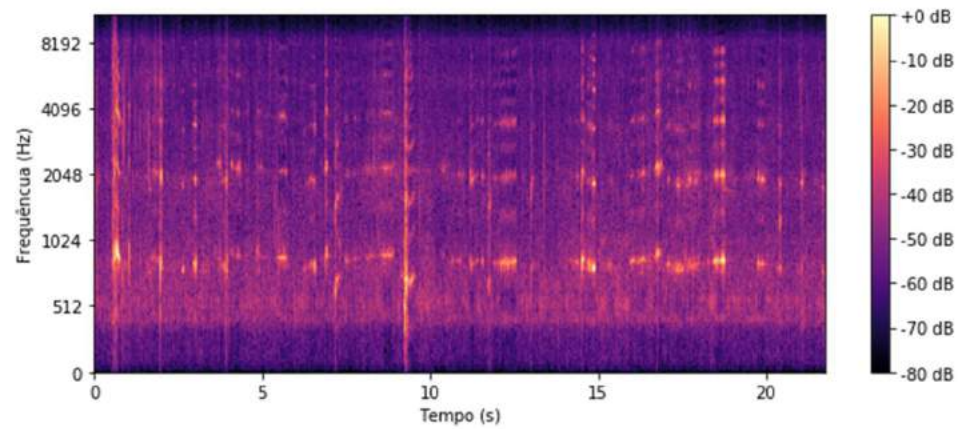
Com o intuito de melhor compreender os dados presentes no *dataset*, analisou-se a forma de onda de um arquivo para cada espécie de mosquito. Também se teve o cuidado de escolher áudios cuja duração fosse maior que alguns segundos, para que a forma de onda não seja apenas um pico. As formas de ondas não trazem informações que permitem a diferenciação das espécies de mosquitos a olho nu, portanto os gráficos estão no APÊNDICE I. Nota-se que há diversos picos ao decorrer do tempo e que em nenhum momento há completo silêncio. Ao reproduzir os arquivos, notou-se que na grande maioria dos casos há ruído ambiente durante a gravação inteira, sendo os picos os únicos momentos em que realmente há gravação do mosquito. Já em outros casos o voo do mosquito sempre está presente na gravação, sendo os picos relativos a distância do mosquito em relação ao microfone.

#### 4.1.2 Análise no domínio da frequência

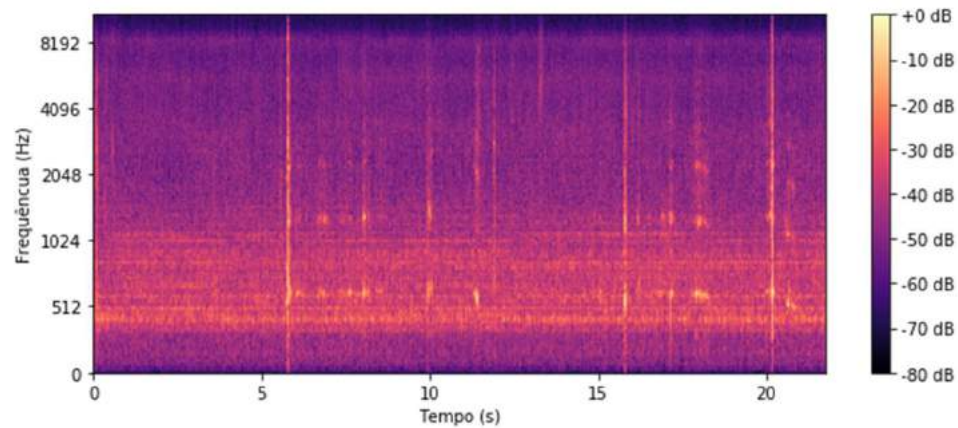
A frequência dos mosquitos está contida entre 200 a 700 Hz, portanto é vantajoso usar uma escala logarítmica para ter maior resolução nessa faixa. Utilizou-se a escala mel pois a mesma é uma escala logarítmica e é nativa na biblioteca de manipulação de áudio utilizada neste trabalho. As Figuras de 16 a 20 apresentam o espectrograma em escala mel de algumas espécies que são vetores de doenças como dengue, febre amarela, febre zika e chikungunya. O espectrograma para as demais espécies encontra-se no APÊNDICE II. Também se extraiu a derivada local do espectrograma utilizando a biblioteca librosa conforme a Figura 21, porém as variações de frequência não são suficientes para que se contenha alguma informação significativa.

Figura 16- Espectrograma do *Aedes aegypti*

Fonte: O autor, 2018.

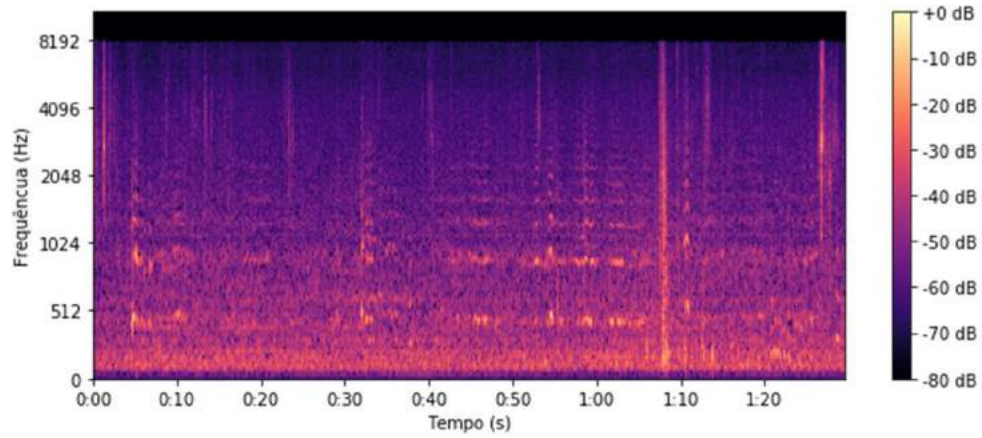
Figura 17 - Espectrograma do *Aedes albopictus*

Fonte: O autor, 2018.

Figura 18 - Espectrograma do *Aedes mediovittatus*

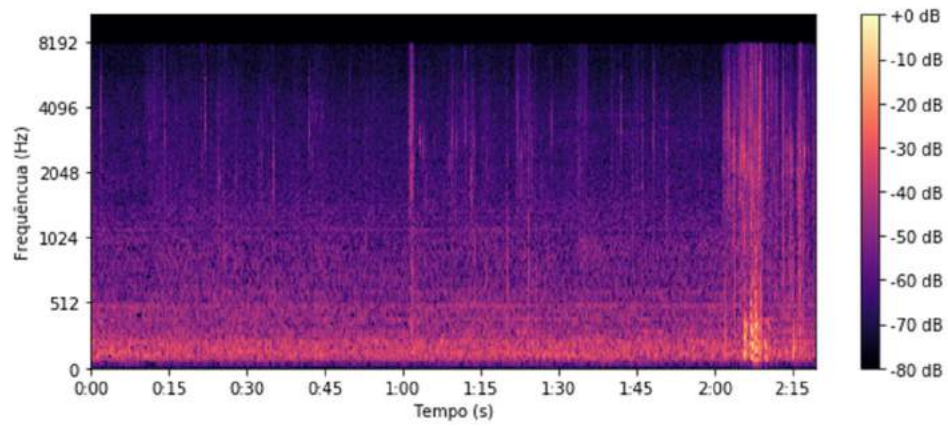
Fonte: O autor, 2018.

Figura 19 - Espectrograma do *Culex pipiens*



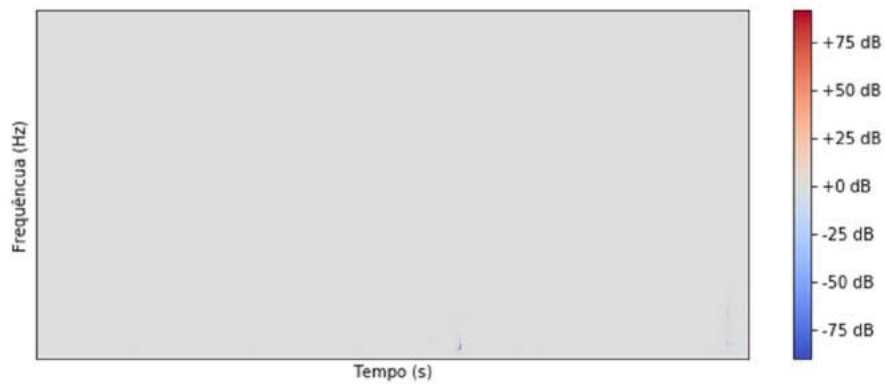
Fonte: O autor, 2018.

Figura 20 - Espectrograma do *Culex quinquefasciatus*



Fonte: O autor, 2018.

Figura 21 - Delta do espectrograma do *Aedes aegypti*



Fonte: O autor, 2018.

## 4.2 NORMALIZAÇÃO DOS ARQUIVOS

Em ML, é fundamental trabalhar com dados normalizados para evitar viés no treinamento dos modelos. Portanto, sabendo que há diferentes formatos de arquivos e diferentes frequências de amostragem, optou-se por converter todos os arquivos para o formato WAV e utilizar a menor frequência de amostragem encontrada nos arquivos originais, que é 8.000 Hz. Visto que a frequência máxima do batimento das asas dos mosquitos encontrado na literatura é de 700 Hz e que 8.000 Hz é muito maior que essa faixa de frequência, o teorema da amostragem de Nyquist-Shannon<sup>5</sup> está sendo devidamente respeitado. Para a conversão do formato de arquivo, utilizou-se o programa de código aberto Audacity e para alterar a frequência de amostragem foi utilizado o utilitário de linha de comando Sox.

Para a remoção de trechos onde não há som de mosquito, abriu-se todos os arquivos usando o Audacity e removeu-os manualmente, sendo que trechos onde o ruído de fundo era quase tão alto quanto o barulho do mosquito também foram removidos. A remoção manual foi necessária pois não se tem um número demasiadamente alto de arquivos e qualquer segmento de áudio é crucial para melhorar o treinamento dos modelos. A duração para cada espécie após a edição encontra-se na Tabela 2. Seguindo esses passos garante-se um certo nível de normalização, com exceção da normalização do volume, o qual não foi modificado.

Tabela 2 - Duração por espécie antes e depois da edição dos arquivos

Espécie	Duração antes da edição (segundos)	Duração após edição (segundos)
<i>Aedes aegypti</i>	1.736,87	1.736,87
<i>Aedes albopictus</i>	966,37	966,37
<i>Aedes mediovittatus</i>	180,00	53,69
<i>Aedes sierrensis</i>	3.337,58	274,01
<i>Anopheles albimanus</i>	1.736,55	901,37
<i>Anopheles arabiensis dongola</i>	1.040,11	850,88
<i>Anopheles arabiensis rufisque</i>	943,76	844,45

<sup>5</sup> De acordo com o teorema, um sinal contínuo no tempo deve ser amostrado utilizando uma frequência de amostragem que seja no mínimo o dobro da maior frequência do sinal original, caso contrário ocorrerá distorções no sinal amostrado as quais são chamadas de *aliasing*.

Espécie	Duração antes da edição (segundos)	Duração após edição (segundos)
<i>Anopheles atroparvus</i>	907,64	833,66
<i>Anopheles dirus</i>	1.733,48	530,35
<i>Anopheles farauti</i>	1.727,52	781,35
<i>Anopheles freeborni</i>	2.397,82	1.237,00
<i>Anopheles gambiae akron</i>	972,98	615,15
<i>Anopheles gambiae kisumu</i>	1.747,59	638,57
<i>Anopheles gambiae rsp</i>	567,66	295,84
<i>Anopheles merus</i>	300,00	205,05
<i>Anopheles minimus</i>	2.923,11	994,16
<i>Anopheles quadriannulatus</i>	1.0178,10	959,33
<i>Anopheles quadrimaculatus</i>	665,71	548,31
<i>Anopheles stephensi</i>	2.909,69	770,22
<i>Culex pipiens</i>	710,37	240,83
<i>Culex quinquefasciatus</i>	758,13	195,07
<i>Culex tarsalis</i>	613,87	231,11
<i>Culiseta incidens</i>	2.971,58	1.244,21

#### 4.3 EXTRAÇÃO DE *FEATURES*

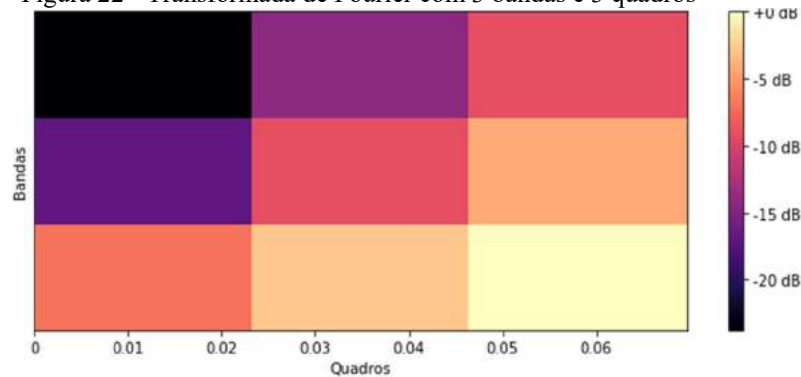
Devido ao fato de um ser humano não ser capaz de identificar diferença entre as espécies ao olhar apenas para o formato de onda, decidiu-se adotar como *feature* somente o espectrograma, onde visivelmente nota-se diferenças entre as espécies analisadas. A escolha do espectrograma como *feature* condiz com o que foi realizado em outros trabalhos. (DIELEMAN; SCHRAUWEN, 2014; PERASCANDOLO; DROSSOS; VIRTANEN, 2017; PICZAK, 2015). Para a extração dessa *feature* deve-se especificar os parâmetros da transformada de Fourier conforme descrito na seção 2.2.1.

Realizada a transformada, tem-se um vetor de duas dimensões onde as linhas representam as frequências analisadas e as colunas o domínio do tempo. As Figuras de 16 a 20 foram obtidas

realizando a transformada com um alto número de bandas e quadros. A Figura 22 apresenta um exemplo simplificado onde realizou-se a transformada com apenas 3 bandas e 3 quadros. Cada pixel dessa imagem contém a informação da intensidade do sinal em decibéis de uma determinada frequência naquele instante de tempo. Ao realizar o espectrograma usando a biblioteca librosa, a mesma retorna valores em uma faixa de -80 a 0 dB conforme a Figura 23. Essa representação é semelhante a uma imagem preto e branco, com a diferença que a escala adotada em imagens preto e branco varia de 0 a 255, porém costuma-se normalizar os valores passados a redes neurais de forma que a entrada varie de 0 a 1. Para normalizar os valores utilizou-se a Equação 10 e o resultado é representado pela Figura 24.

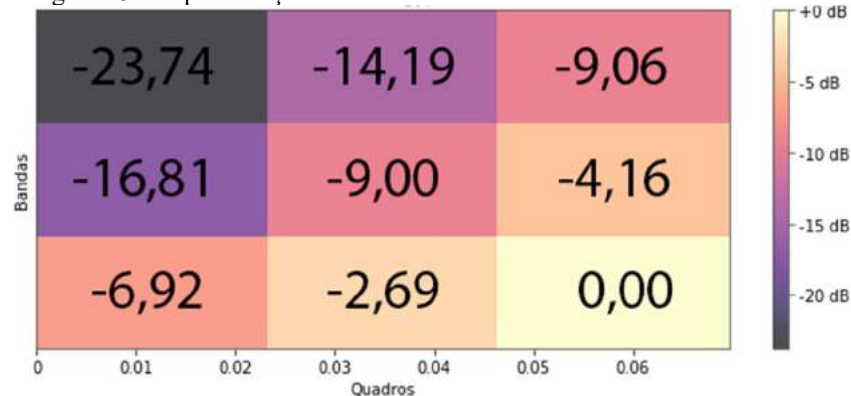
$$y = \frac{x}{80} + 1 \quad (10)$$

Figura 22 - Transformada de Fourier com 3 bandas e 3 quadros

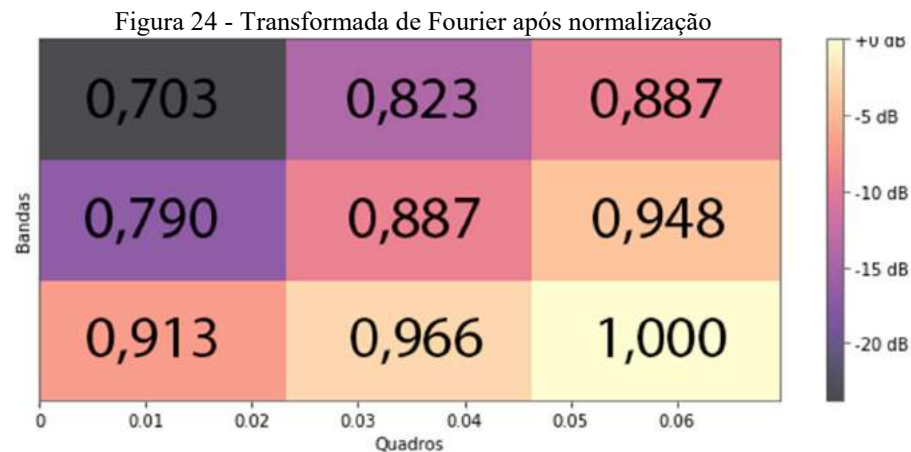


Fonte: O autor, 2018

Figura 23 - Representação em decibéis da transformada de Fourier



Fonte: O autor, 2018



Fonte: O autor, 2018

Portanto, após realizar a transformada de Fourier, transforma-se o problema de classificação de áudio em um problema de classificação de imagens, sendo então, possível utilizar técnicas de classificação de imagens (principalmente CNNs) para realizar a classificação de áudios. Como o *dataset* não contém muitos dados, optou-se por sobrepor os áudios em 50% para obter mais amostras. Seguindo o que foi realizado por Cakir (2017), utilizou-se os arquivos originais e arquivos modificados por um filtro passa-faixa de 100 a 1.000 Hz.

#### 4.4 MODELO DE MACHINE LEARNING

Os modelos foram construídos utilizando a biblioteca Keras, que é uma API de alto nível para criação de redes neurais utilizando TensorFlow e Python. Para a validação cruzada usou a *scikit-learn* (PEDREGOSA, 2011). Criou-se diferentes modelos com diferentes estratégias a fim de comparar os resultados dos mesmos. Para que não ocorresse *overfitting* durante o processo de treinamento e validação escolheu-se o número de iterações (epochs) analisando o gráfico da acurácia e erro do conjunto de treinamento e validação ao longo das interações. As métricas de desempenho, apresentadas na seção 2.6, foram obtidas utilizando validação cruzada com 10-*folds* e não se separou dados para teste independente da validação cruzada devido à escassez de dados disponíveis para treinamento.

#### 4.4.1 Classificador multiclasse

Primeiro considerou-se cada estirpe como uma classe individual e, portanto, criou-se um modelo com o intuito de classificar o som entre as 23 possíveis classes. Devido ao fato do trabalho se tratar da identificação da espécie *Aedes aegypti* em específico não se aprofundou o estudo nessa estratégia multiclasse. Porém é interessante verificar o desempenho desse modelo e principalmente sua matriz de confusão pois ambos permitem entender melhor os desafios ao tentar classificar espécies de mosquito e permitem a comparação com o resultado obtido no projeto Abuzz que não utiliza ML.

#### 4.4.2 Classificador binário

Nessa estratégia criou-se um modelo binário onde uma classe é o *Aedes aegypti* e a outra classe contém todas as demais espécies. Essa rede é extremamente desbalanceada, porém é a abordagem mais simples para realizar a classificação.

#### 4.4.3 Ensemble de classificadores binários

O último modelo desenvolvido consiste em um *ensemble* composto de 22 classificadores binários. Esses classificadores binários tem o *Aedes aegypti* como classe comum, porém a outra classe é uma espécie de mosquito diferente para cada classificador. Portanto, quando o *ensemble* for confrontado com um áudio do *Aedes aegypti*, espera-se que a maioria dos classificadores binários irá classificar o mesmo corretamente. Por outro lado, quando for outra espécie cada classificador irá apresentar uma resposta diferente. Ao invés de simplesmente fazer uma média ou votação majoritária dos classificadores binários, o *ensemble* é treinado utilizando uma pequena rede neural.

Primeiro treinou-se os 22 classificadores utilizando validação cruzada para obter as métricas de desempenho individuais para cada classificador, depois treinou-se cada classificador utilizando como treinamento toda a base de dados. Por fim realizou-se validação cruzada na rede neural do *ensemble*, iniciando com o menor número de neurônios possível e adicionando neurônios até que o resultado não melhorasse.



## 5 RESULTADOS E DISCUSSÕES

Nessa seção encontra-se os resultados dos diferentes modelos desenvolvidos, bem como uma análise das métricas obtidas.

Em um cenário ideal, o modelo desenvolvido seria capaz de identificar a espécie do mosquito com métricas iguais a 100%, porém não sendo possível alcançar tal modelo deve-se priorizar um meio termo entre precisão e *recall*. Tratando-se do contexto de identificação do *Aedes aegypti* é preferível realizar uma classificação do tipo falso positivo a uma falsa negativa, ou seja, é preferível identificar o mosquito como sendo *Aedes aegypti* mesmo que o mesmo não seja do que afirmar que não é *Aedes aegypti* quando o mesmo é. Portanto, como o *recall* mede o número de instâncias positivas que foram corretamente identificadas pelo classificador, é preferível ter um *recall* alto a uma precisão alta.

Devido ao fato de ter sido utilizado validação cruzada, os gráficos gerados para a análise das métricas são gráficos de caixa. Esse tipo de gráfico contém uma linha laranja que é a mediana, apresenta limite inferior e superior que representam os dois valores extremos do conjunto de dados, e também o primeiro e terceiro quartil (WILLIAMSON, 1989). Não se considerou valores atípicos (em inglês *outliers*), pois todos os valores foram métricas de desempenho da validação cruzada. Caso haja uma grande discrepância entre o valor máximo e mínimo significa apenas que o modelo se comportou de maneira diferente em cada uma das *folds*.

### 5.1 MODELO DAS REDES NEURAIS CONVOLUCIONAIS

Ao iniciar a avaliação de qual configuração de rede melhor se adapta a resolução desse tipo de problema e a influência dos parâmetros da transformada de Fourier, percebeu-se que o mesmo modelo usado para realizar a classificação binária entre *Aedes aegypti* e outra espécie qualquer apresenta bons resultados tanto para a classificação multiclasse quanto para a classificação binária entre *Aedes aegypti* versus todas as demais espécies. Portanto, primeiro buscou-se encontrar um conjunto de parâmetros e uma arquitetura de modelo base que apresentasse resultados satisfatórios para a classificação binária do *Aedes aegypti* versus outra espécie, para então testar diferentes valores de parâmetros e replicar esse modelo para as estratégias descritas na metodologia. A espécie utilizada para o treinamento inicial foi a *Anopheles freeborni*, pois possui 1.237s de

gravações, o que é próximo dos 1.736,87s do *Aedes aegypti* e, portanto, essas duas classes não estão completamente desbalanceadas como é o caso de outras espécies.

O sumário da arquitetura escolhida é exibida na Figura 25. Foram testadas demais arquiteturas sendo possível atingir boas métricas com uma arquitetura simples. Escolheu-se essa arquitetura pois a mesma apresenta um bom desempenho e treinando-a corretamente não ocorre *overfitting*. Na primeira camada escolheu-se utilizar uma matriz com uma altura que é quase metade da altura da rede a fim de evitar os problemas descritos por ROTHMANN (2018), onde uma rede CNN iria aprender padrões locais que fazem sentido quando aplicados em imagens, mas não o fazem quando aplicados a um espectrograma. Nesse sumário da arquitetura também consta que para cada modelo serão calculados 38,722 parâmetros durante a etapa de treinamento com o intuito de encontrar o menor erro. A rede, no caso de um classificador binário, é composta por essas camadas em sequência:

- Camada convolutiva com 32 filtros, *kernel* 20×5 e ativação ReLU.
- Camada de *pooling* 2×2 e *stride* 1.
- Camada de *droupout* 0,2.
- Camada convolutiva com 32 filtros, *kernel* 8×4 e ativação ReLU.
- Camada de *pooling* com filtro 2×2 e *stride* 1.
- Camada totalmente conectada com 1.334 neurônios e ativação ReLU.
- Camada totalmente conectada com 2 neurônios a ativação Sigmóide.

Figura 25 - Modelo de CNN com o melhor desempenho, nesse caso por ter 2 neurônios na saída é um classificador binário

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 41, 37, 32)	3232
max_pooling2d_1 (MaxPooling2)	(None, 20, 18, 32)	0
dropout_1 (Dropout)	(None, 20, 18, 32)	0
conv2d_2 (Conv2D)	(None, 13, 15, 32)	32800
max_pooling2d_2 (MaxPooling2)	(None, 6, 7, 32)	0
flatten_1 (Flatten)	(None, 1344)	0
dense_1 (Dense)	(None, 2)	2690
Total params: 38,722		
Trainable params: 38,722		
Non-trainable params: 0		

Fonte: O autor, 2018.

Nas Figuras 26 a 29 possível observar o impacto ao alterar o valor dos parâmetros utilizados na transformada de Fourier na acurácia, precisão, *recall* e *F1-score*. Os modelos testados usaram os valores de parâmetros da Tabela 3, sendo o termo “modelo  $n$ ” uma referência ao modelo que foi avaliado utilizando o conjunto de parâmetros  $n$ . Obteve-se métricas de desempenho elevadas já que os piores casos foram aqueles em que a medida de desempenho foi 0,9.

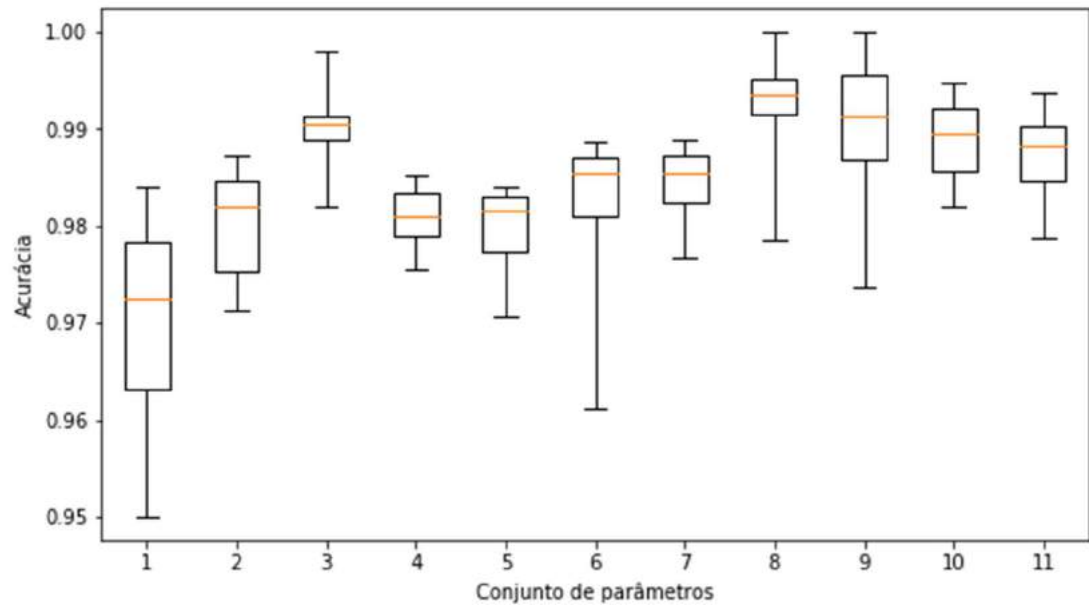
Analisando os modelos de 1 a 4, percebe-se que o melhor desempenho foi obtido quando se manteve o número de bandas em 60. Já para o número de quadros considerou-se os modelos 4, 5 e 6 onde se teve desempenho melhor em um indicador ao custo de um desempenho inferior em outro. O parâmetro *hop length*, por sua vez, foi variado nos modelos de 6 a 9 e apresentou melhores resultados no modelo 8. O último parâmetro referente ao tamanho da janela foi alterado nos modelos 9, 10 e 11 sendo esses dois últimos superiores ao modelo 9.

Analisando os gráficos e priorizando-se as métricas de *recall* e *F1-score* percebe-se que os modelos 3 e 8 apresentaram os melhores desempenhos.

Tabela 3 – Relação dos modelos e seus conjuntos de parâmetros

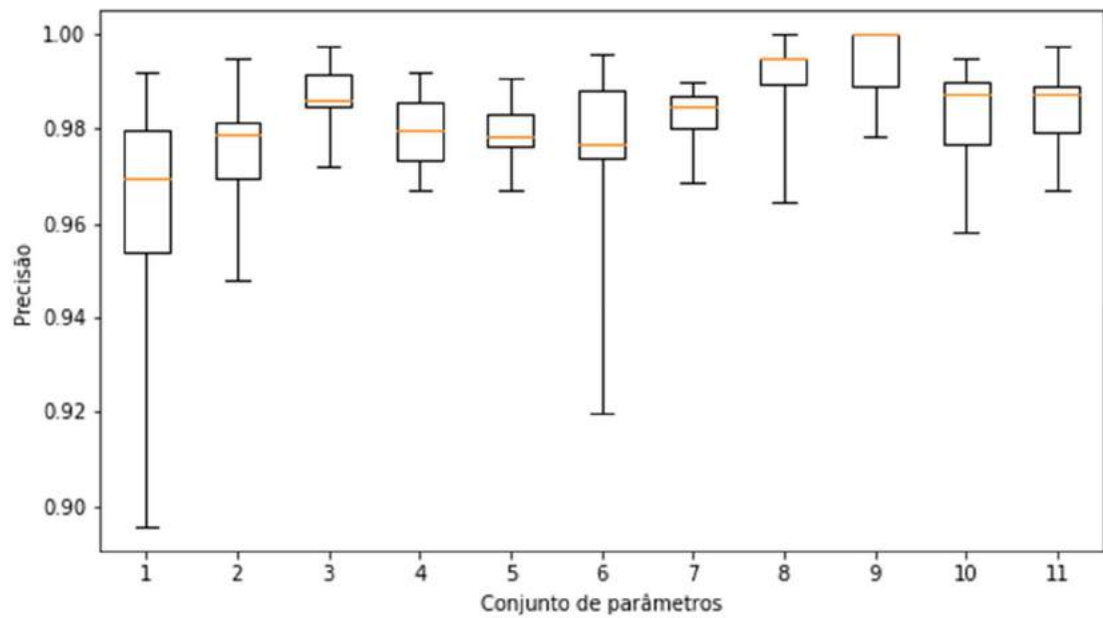
Conjunto	Número de bandas	Número de quadros	<i>Hop length</i>	Tamanho da janela
1	20	40	128	1024
2	40	40	128	1024
3	60	40	128	1024
4	80	40	128	1024
5	60	20	128	1024
6	60	60	128	1024
7	60	40	64	1024
8	60	40	256	1024
9	60	40	512	1024
10	60	40	128	512
11	60	40	128	2048

Figura 26 - Acurácia do classificador *Aedes aegypti* vs. *Anopheles freeborni* variando-se os parâmetros



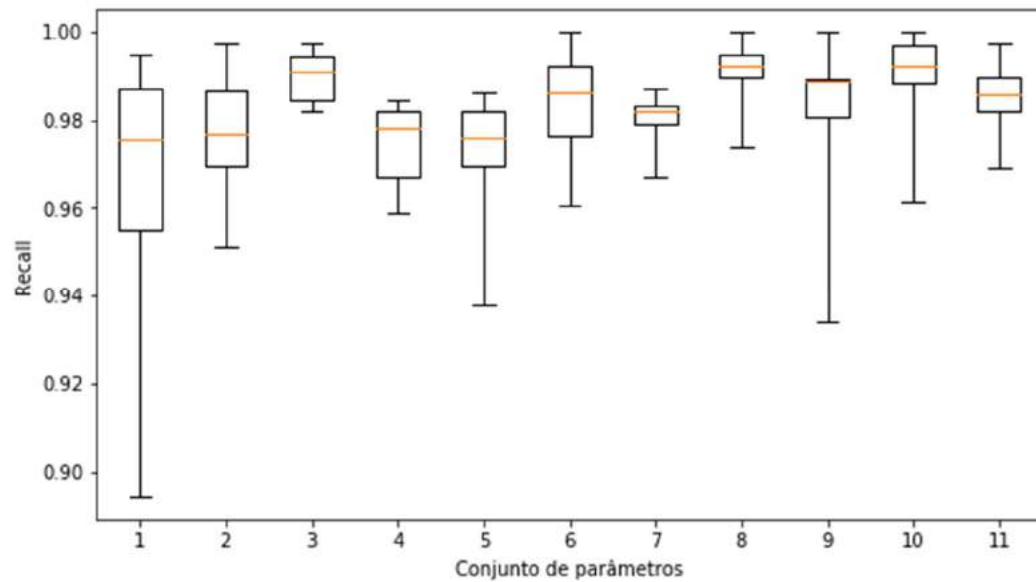
Fonte: O autor, 2018.

Figura 27 - Precisão do classificador *Aedes aegypti* vs. *Anopheles freeborni* variando-se os parâmetros



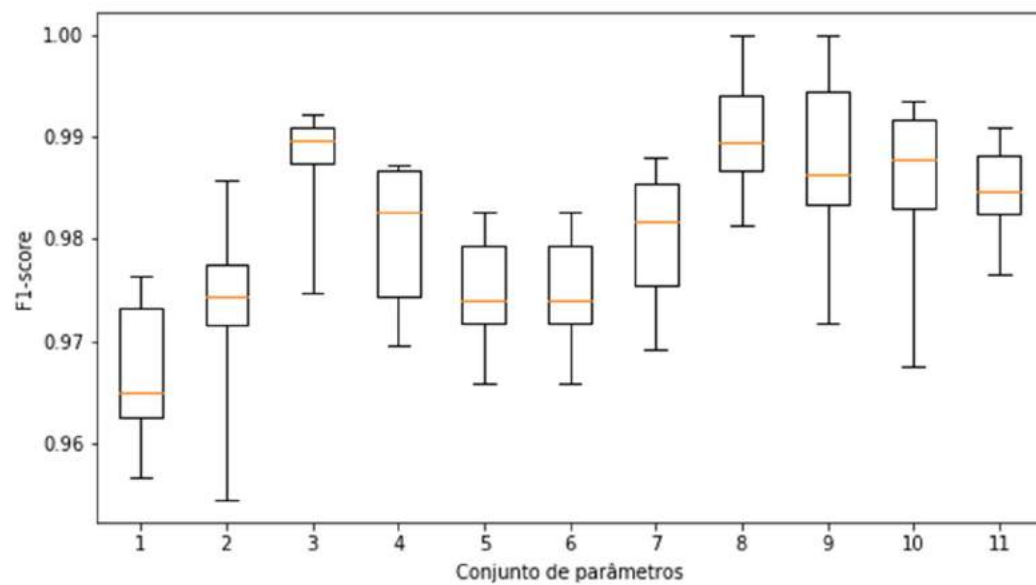
Fonte: O autor, 2018.

Figura 28 - *Recall* do classificador *Aedes aegypti* vs. *Anopheles freeborni* variando-se os parâmetros



Fonte: O autor, 2018.

Figura 29 - *F1-Score* do classificador *Aedes aegypti* vs. *Anopheles freeborni* variando-se os parâmetros



Fonte: O autor, 2018.

A fim de verificar se o modelo estava utilizando *features* fora da faixa de frequência do mosquito, treinou-se um modelo onde os arquivos receberam um filtro passa-faixa na faixa de 100 a 1.000 Hz. Esse modelo não demonstrou diferença significativa quando comparado com o que foi treinado utilizando todo o espectro de frequência. O ajuste de sobreposição se aumentado para além de 50% não teve influência significativa nas métricas, porém ao não utilizar sobreposição teve-se

uma queda no desempenho. Portanto sugere-se utilizar sobreposição para esse tipo de análise onde o segmento de áudio é curto.

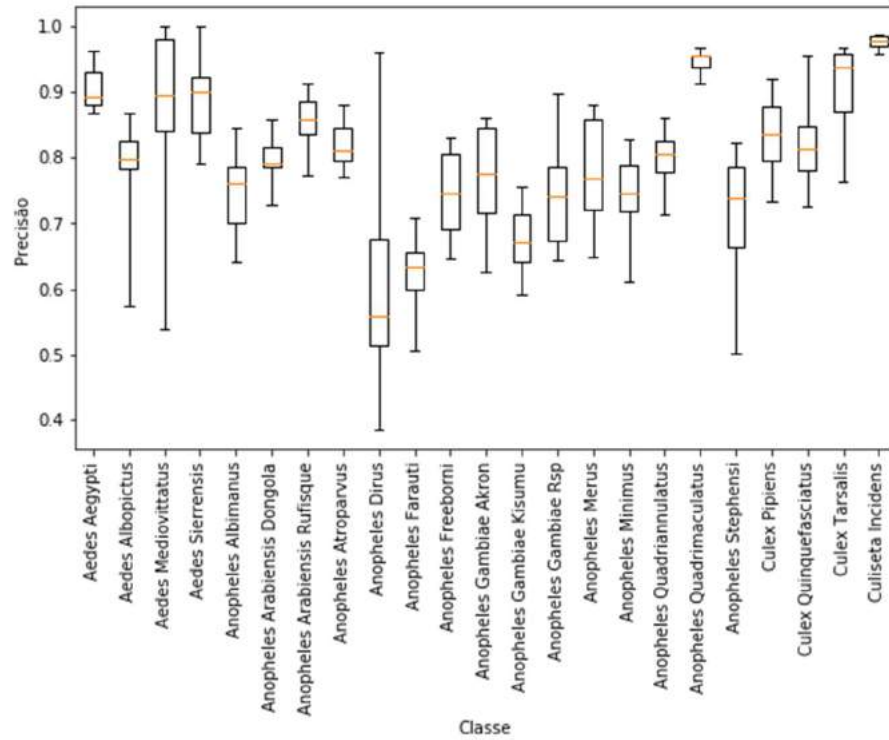
## 5.2 DESEMPENHO CLASSIFICADOR MULTICLASSE

A arquitetura de rede utilizada foi similar à da Figura 25, porém alterou-se a última camada para que a mesma possua 23 neurônios e sua função de ativação para *softmax*. As Figuras 30 a 32 apresentam os indicadores de desempenho utilizando o conjunto de parâmetros 3 da Tabela 3, já as Figuras 33 a 35 apresentam as métricas utilizando o conjunto de parâmetros 8. Na Figura 36 encontra-se a acurácia dos dois modelos. A matriz de confusão do modelo 3 encontra-se no APÊNDICE III. O classificador conseguiu identificar corretamente o *Aedes aegypti*, sendo os piores resultados quando se tenta classificar *Aedes aegypti*, *Aedes albopictus* e *Aedes mediovittatus*. No caso das espécies *Anopheles arabiensis* e *Anopheles gambiae* que possuem diferentes estirpes o modelo não teve dificuldade em diferenciar entre os mesmos, confirmando os resultados encontrados na pesquisa de Offenhauser e Kahn (1949) porém discordando parcialmente com o que se encontrou em Mukundarajan *et al.* (2017), visto que neste a espécie *Anopheles gambiae* apresentou a mesma frequência fundamental, mas a *Anopheles arabiensis* não. A média macro dos indicadores de desempenho dos classificadores encontra-se na Tabela 4.

Tabela 4 - Macro média dos indicadores de desempenho multiclasse

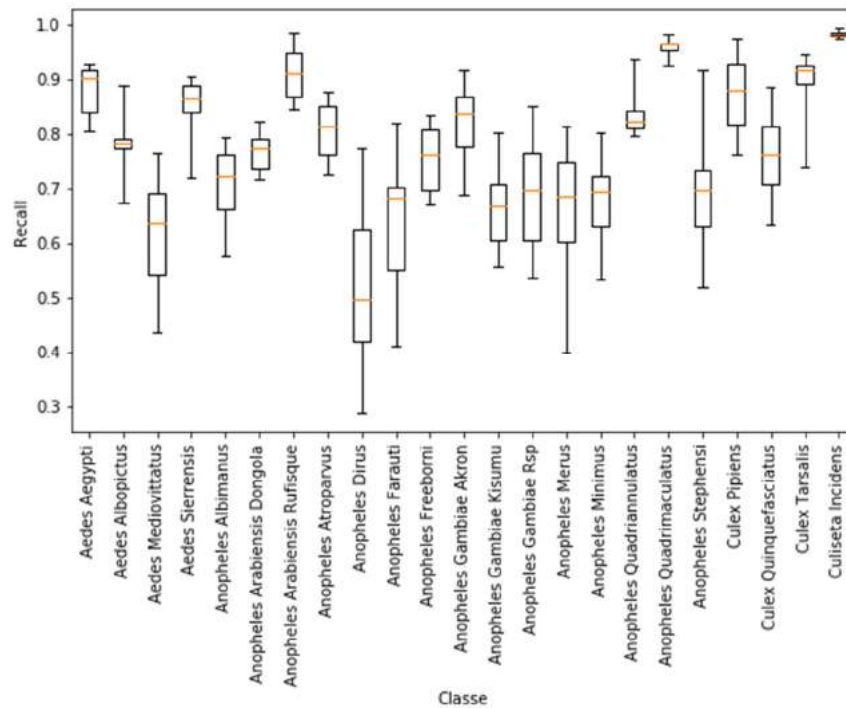
Modelo	Acurácia (%)	Precisão (%)	Recall (%)	F1-score (%)
3	79,00 ±1,32	79,76 ±1,27	77,17 ±1,32	78,01 ±1,45
8	80,47 ±1,32	81,30 ±0,97	79,06 ±1,41	79,28 ±1,37

Figura 30 - Precisão do modelo multiclasse com conjunto de parâmetros 3



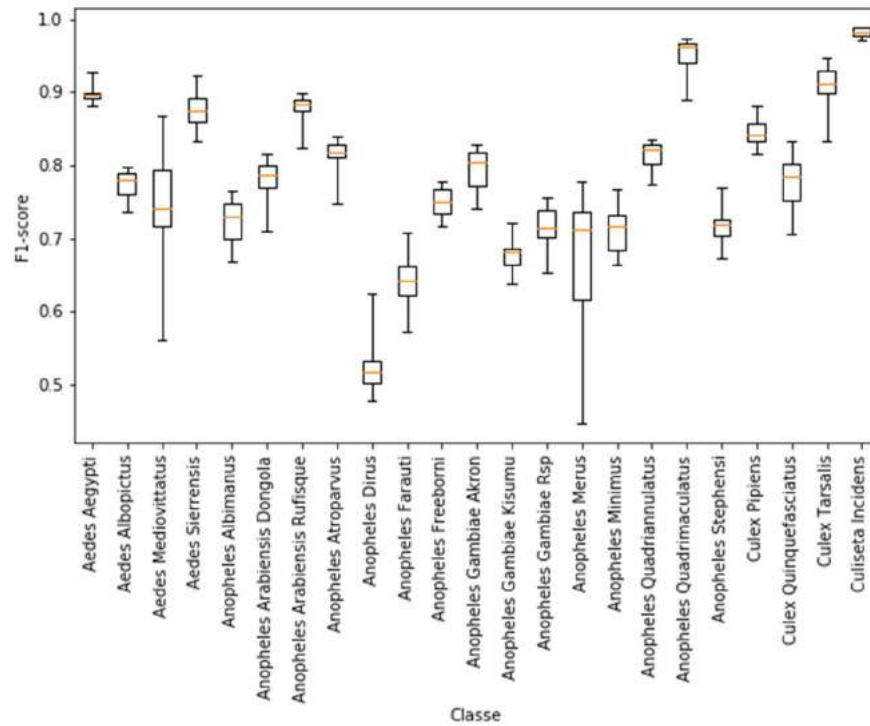
Fonte: O autor, 2018.

Figura 31 - Recall do modelo multiclasse com conjunto de parâmetros 3



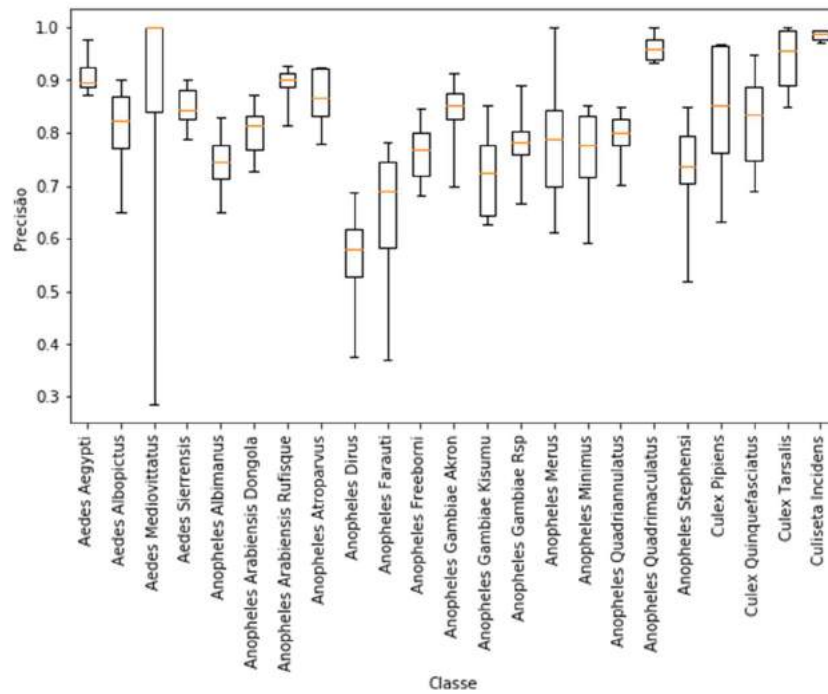
Fonte: O autor, 2018.

Figura 32 – *F1-score* do modelo multiclasse com conjunto de parâmetros 3



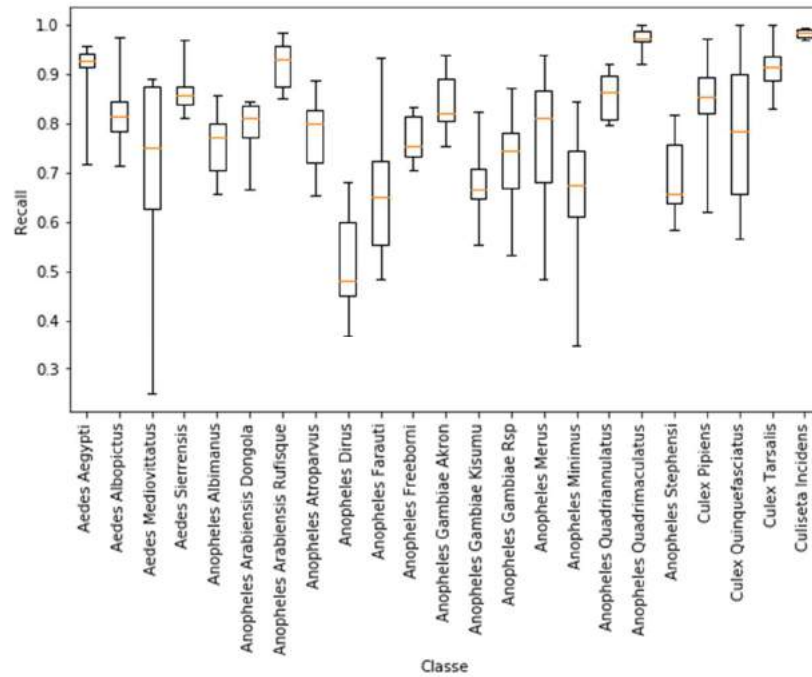
Fonte: O autor, 2018.

Figura 33 - Precisão do modelo multiclasse com conjunto de parâmetros 8

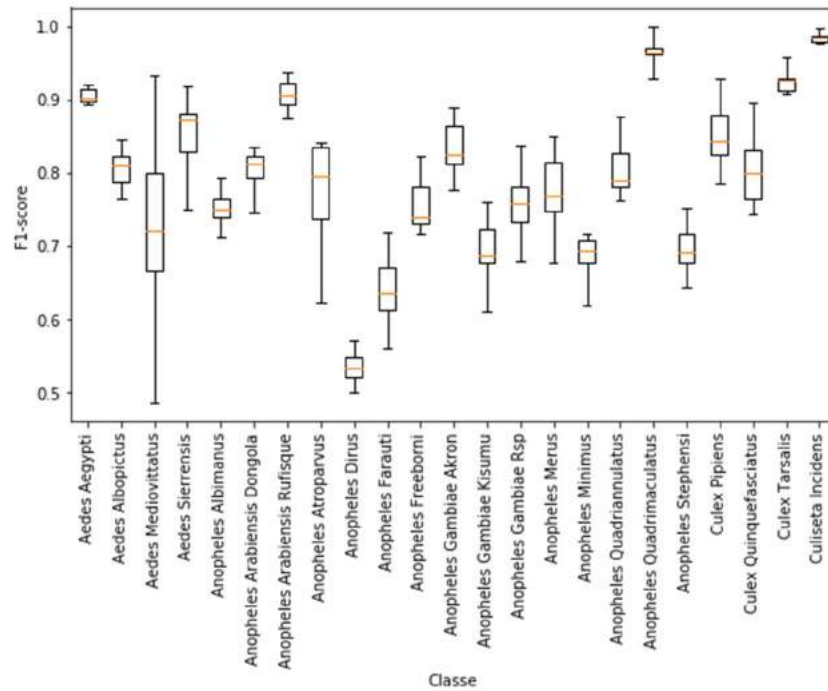


Fonte: O autor, 2018.



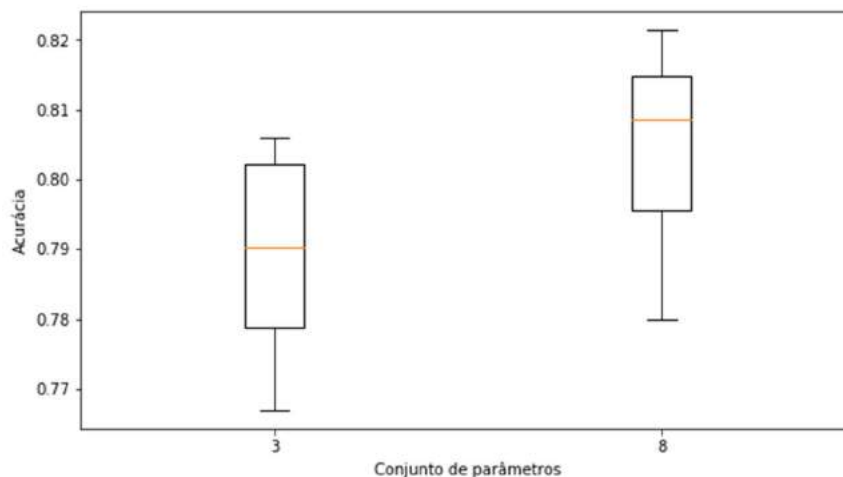
Figura 34 - *Recall* do modelo multiclasse com conjunto de parâmetros 8

Fonte: O autor, 2018.

Figura 35 - *F1-score* do modelo multiclasse com conjunto de parâmetros 8

Fonte: O autor, 2018.

Figura 36 - Acurácia dos modelos multiclass



Fonte: O autor, 2018.

A acurácia dos dois modelos supera aquela encontrada em Mukundarajan *et al.* (2017), que foi de 65% utilizando metadados geográficos e se aproxima da acurácia encontrada em Chen (2018) que foi de 80% considerando apenas 10 espécies de mosquitos e também utilizando metadados geográficos.

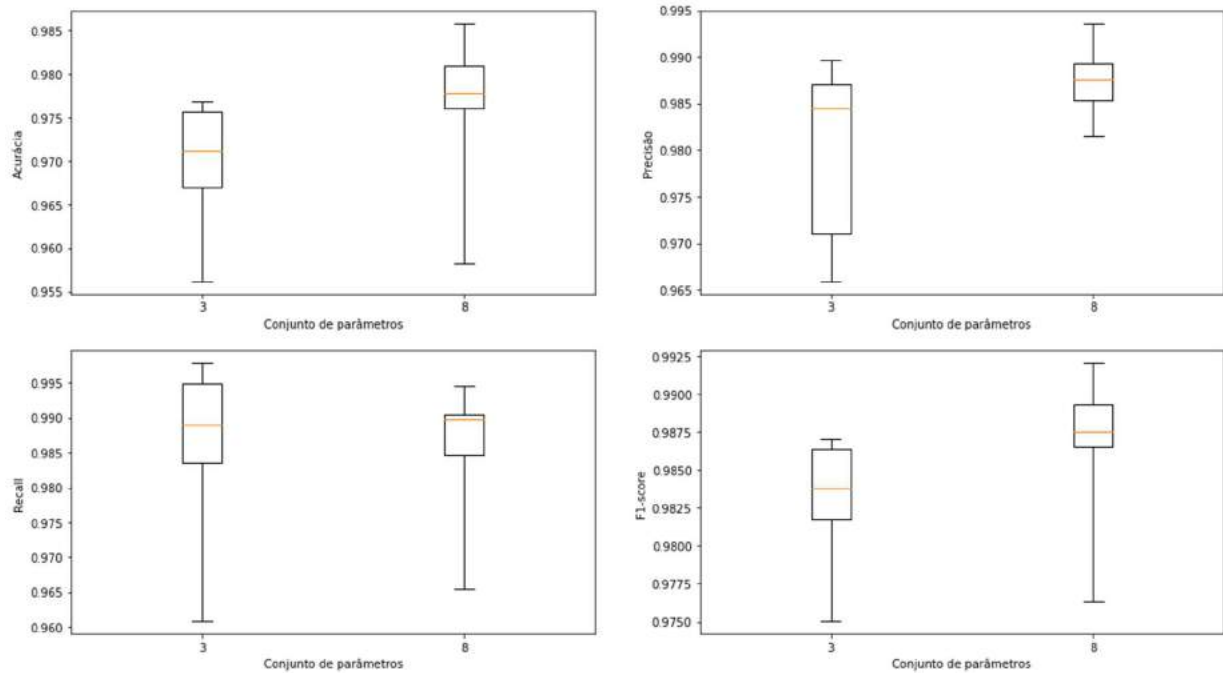
### 5.3 DESEMPENHO CLASSIFICADOR BINÁRIO

Esse classificador está desbalanceado, pois apenas 11% dos dados são da espécie *Aedes aegypti* especificamente, têm-se 5.537 quadros contra 44.713 das demais espécies. Utilizou-se dois conjuntos de parâmetros, 3 e 8, que foram os que demonstraram melhor desempenho no classificador *Aedes aegypti* vs. *Anopheles freeborni*. As médias dos indicadores e seus respectivos desvios padrões encontram-se na Tabela 5 e são representados utilizando gráfico de caixa na Figura 37.

Tabela 5 - Indicadores de desempenho classificador binário *Aedes aegypti* vs. demais espécies

Modelo	Acurácia média (%)	Desvio padrão acurácia (%)	Precisão média (%)	Desvio padrão precisão (%)	Recall médio (%)	Desvio padrão recall (%)	F1-score médio (%)	Desvio padrão F1- score (%)
3	97,12	0,65	98,45	0,96	98,89	1,10	98,38	0,37
8	97,78	0,73	98,75	0,34	98,97	0,81	98,75	0,42

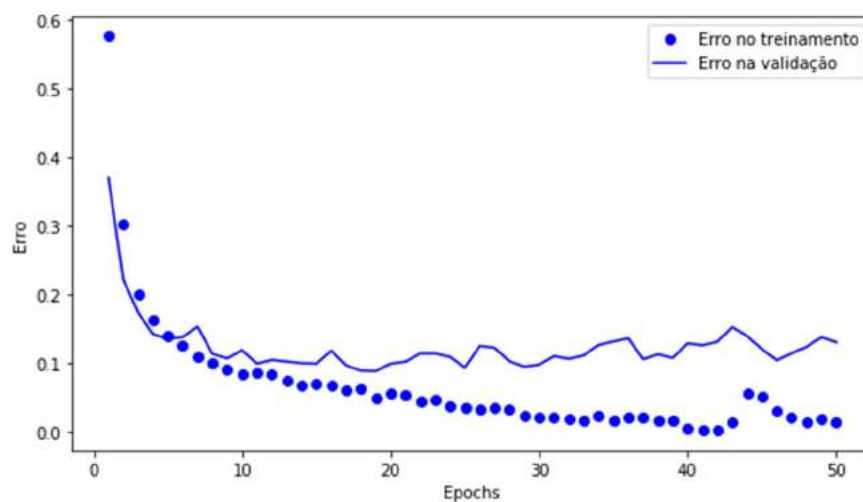
Figura 37 - Indicadores de desempenho classificador binário *Aedes aegypti* vs. demais espécies



Fonte: O autor, 2018.

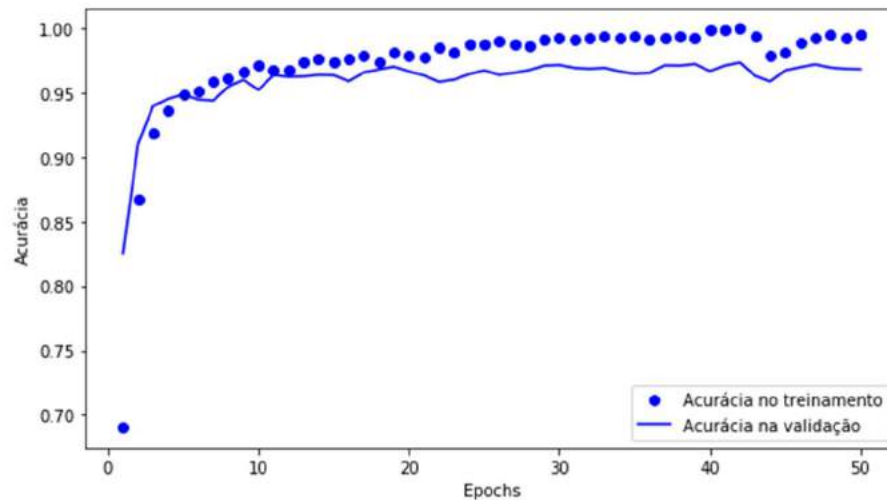
O modelo 8 obteve média superior em todos os indicadores de desempenho, porém tendo desvio padrão maior na acurácia e no *F1-score*. As Figuras 38 e 39 apresentam os gráficos do erro e da validação ao longo de cada iteração durante o treinamento. Com base nesses gráficos, decidiu-se utilizar apenas 10 iterações com tamanho de *batch* igual a 32.

Figura 38 - Erro no conjunto de treinamento e validação a cada iteração



Fonte: O autor, 2018.

Figura 39 - Acurácia no conjunto de treinamento e validação a cada iteração

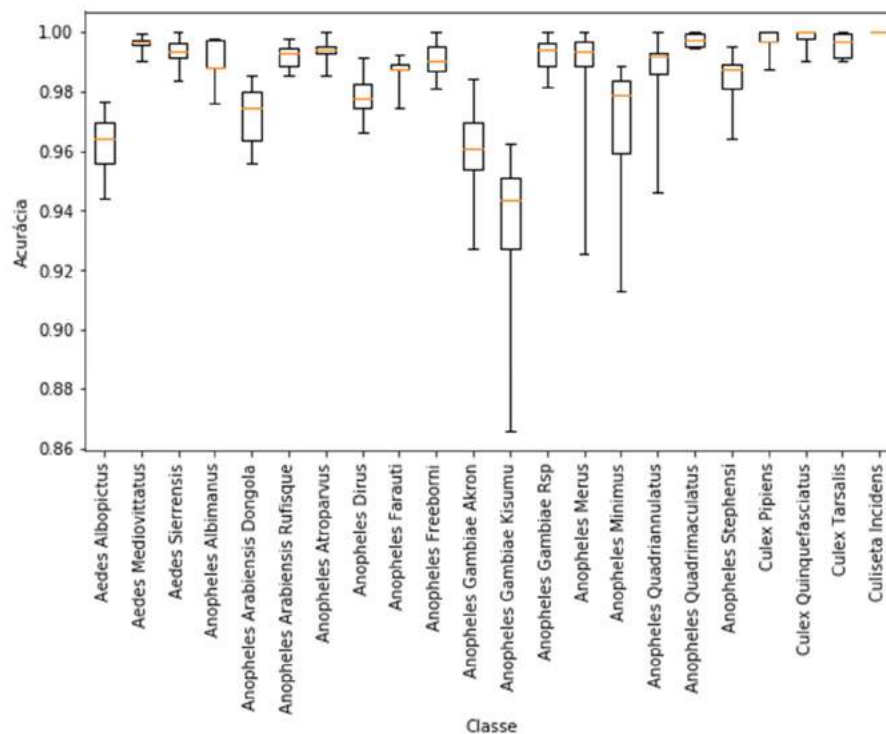


Fonte: O autor, 2018.

### 5.3 DESEMPENHO *ENSEMBLE* DE CLASSIFICADORES BINÁRIOS

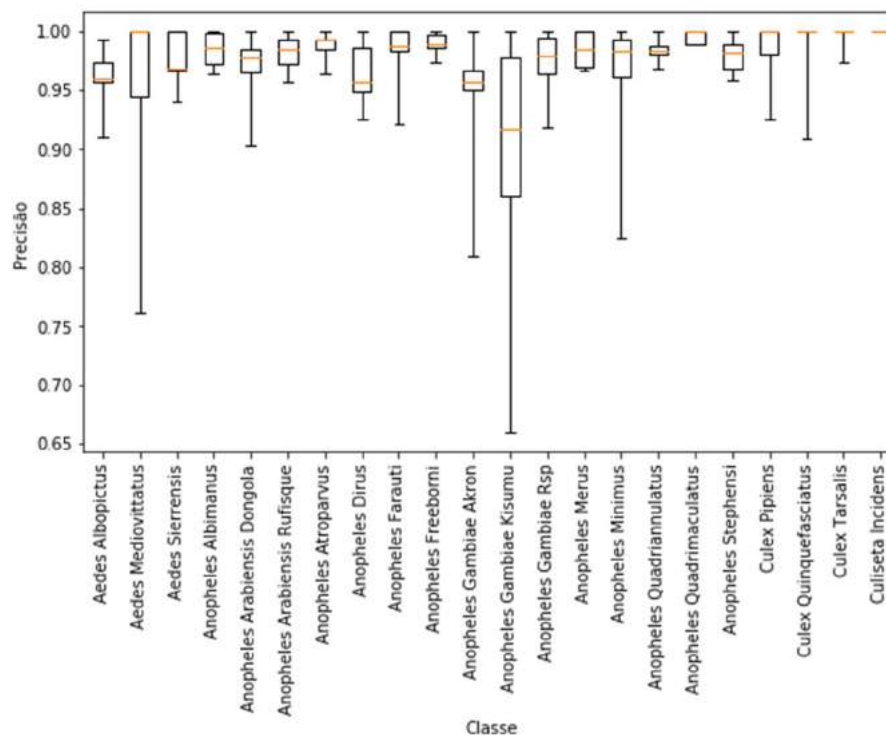
Considerando-se que os melhores parâmetros utilizados durante o espectrograma mel foram obtidos utilizando o conjunto de parâmetros 8, realizou-se o treinamento dos 22 modelos binários nessa configuração. Os indicadores de desempenho individuais de cada espécie são exibidos nas Figuras 40 a 43. Importante ressaltar que para os gêneros *Culex* e *Culiseta* obtiveram-se altos indicadores de desempenho, inclusive chegando a 100% de acurácia para o *Culiseta incidens*. Possivelmente devido ao fato de esses mosquitos terem sua frequência fundamental em uma faixa de 200 a 400 Hz enquanto a do *Aedes aegypti* é de 450 a 600 Hz.

Figura 40 - Acurácia dos classificadores binários

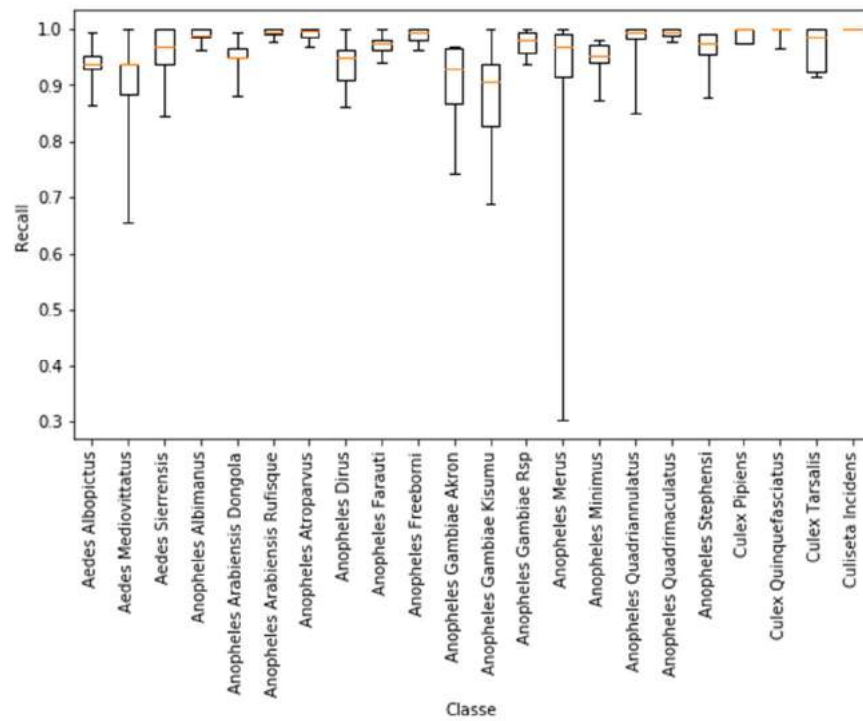


Fonte: O autor, 2018.

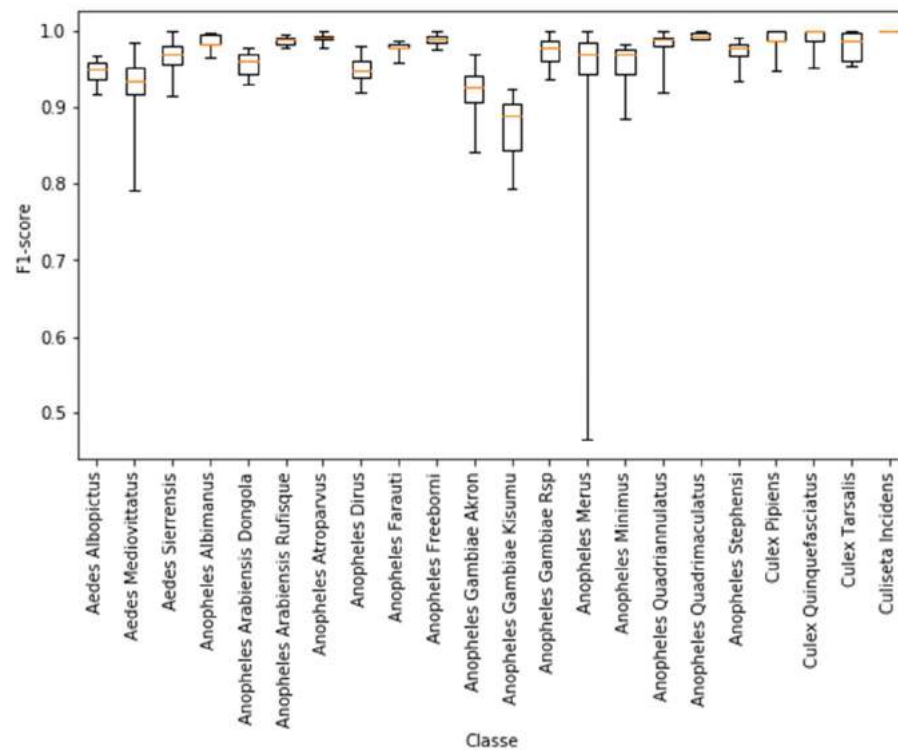
Figura 41 - Precisão classificadores binários



Fonte: O autor, 2018.

Figura 42 - *Recall* classificadores binários

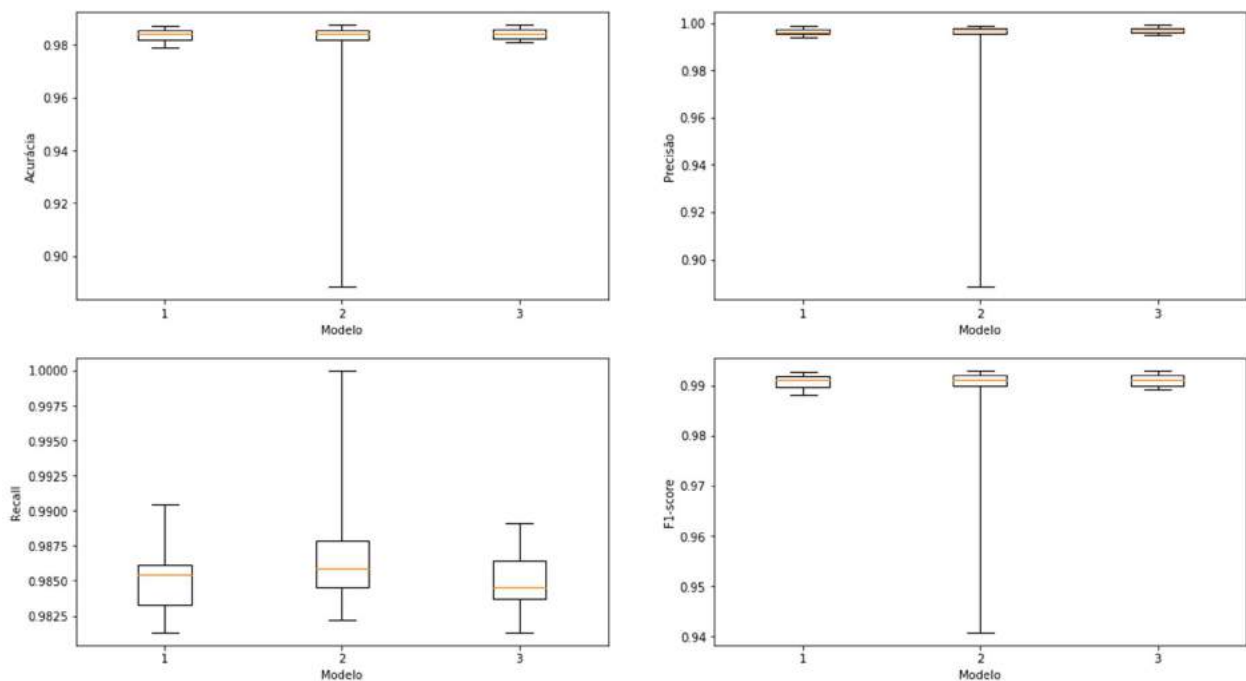
Fonte: O autor, 2018.

Figura 43 - *F1-score* classificadores binários

Fonte: O autor, 2018.

Para a elaboração do *ensemble*, primeiro testou-se o modelo mais simples, que consiste em apenas 22 neurônios de entrada conectados a um único neurônio de saída. Depois resolveu-se adicionar uma camada intermediária com 2 neurônios e aumentar a mesma progressivamente até que os indicadores de desempenho decaíssem. Contudo, conforme pode ser visto na Figura 44, a diferença entre as configurações foram mínimas. O modelo 1 consiste em 22 neurônios na primeira camada e 1 na última, no modelo 2 acrescenta-se uma camada intermediária com 2 neurônios, e o modelo 3 tem três neurônios na camada intermediária.

Figura 44 - Três configurações de ensemble testadas

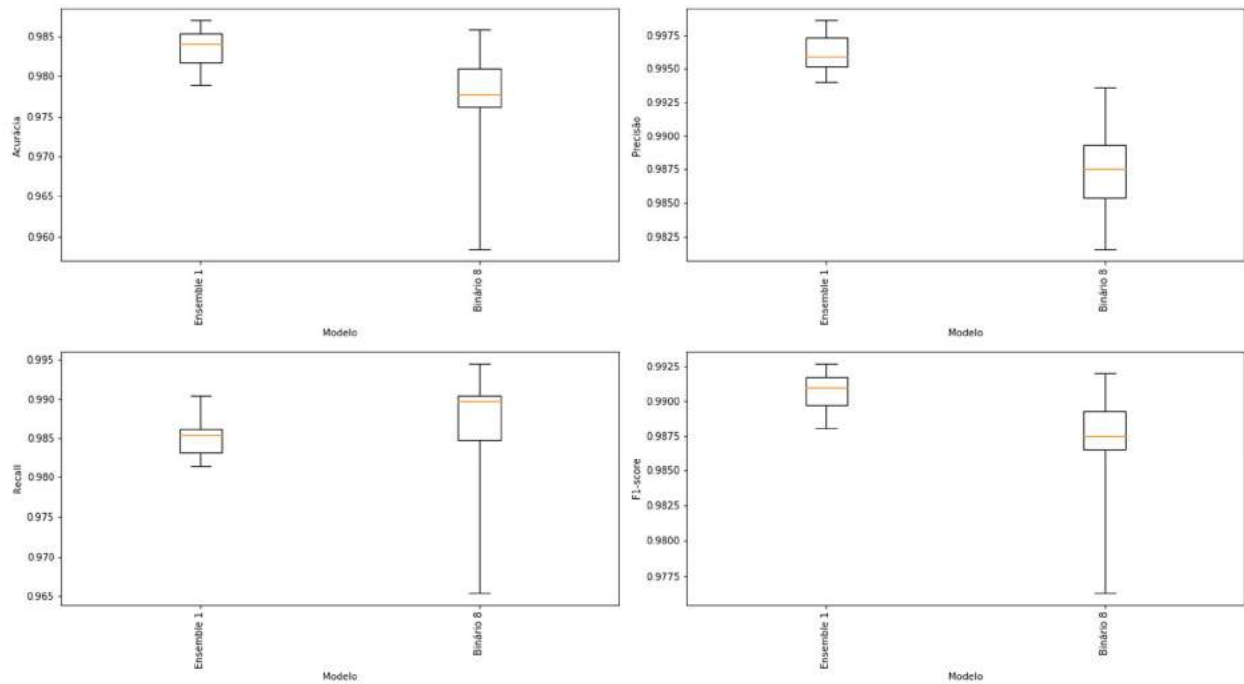


Fonte: O autor, 2018.

Na Figura 45 tem-se a comparação entre os dois classificadores que apresentaram melhor desempenho: o classificador *ensemble* utilizando 22 neurônios na primeira camada e 1 na última e o classificador binário empregando o conjunto de parâmetros 8. Além da diferença de desempenho entre os classificadores, também há a diferença computacional que é maior no *ensemble*, pois o mesmo consiste em 22 CNNs mais o modelo do *ensemble* em si. Portanto, se torna um modelo mais custoso para realizar a classificação e também para treinar o mesmo. Também deve-se ter em mente que os classificadores binários que compõe o *ensemble* foram treinados utilizando todo o

*dataset*, portanto, embora o *ensemble* não conheça os dados utilizados em sua validação, os modelos que o compõe conhecem.

Figura 45 - Comparação entre *ensemble* e classificador binário



Fonte: O autor, 2018.



## 6 CONCLUSÕES

O presente trabalho questionando se seria possível realizar o mapeamento geoespacial e temporal de mosquitos utilizando aparelhos celulares. Ao pesquisar sobre o assunto na literatura, deparou-se com um ramo não muito explorado, porém, identificou-se um estudo prévio no qual se provou a possibilidade de realizar gravações com qualidade suficiente para identificação dos mosquitos usando aparelhos celulares. Decidiu-se utilizar o mesmo *dataset* desse estudo, aplicando-se técnicas de *Machine Learning* com o objetivo de identificar a espécie *Aedes aegypti* em específico, devido aos problemas que o mesmo traz em território nacional. Desenvolveu-se três estratégias utilizando diferentes classificadores que utilizam redes neurais convolucionais com o objetivo de comparar o resultado dos mesmos e entender melhor o problema. Ao realizar um filtro passa-faixa nas gravações, não se notou diferença significativa nos resultados, o mesmo acontece quando se usa sobreposição entre os segmentos de mais de 50%.

O classificador multiclasse apresentou o pior desempenho, com acurácia de  $80,00\% \pm 1,3$ . Porém, seu desempenho foi comparável ao existente na literatura, sendo que em outras pesquisas utilizaram-se metadados geográficos. O classificador binário entre *Aedes aegypti* vs. demais classes apresentou o segundo melhor desempenho com acurácia de  $97,78\% \pm 0,73$ . O último classificador, que consiste em um *ensemble* de diversos classificadores binários, apresentou acurácia de  $98,34\% \pm 0,27$ . Os classificadores binários individuais apresentaram métricas de desempenho em sua grande maioria acima de 90% e analisando o número de amostras de cada espécie percebe-se que as espécies que obtiveram desempenho inferiores não o tiveram devido à falta de dados. Portanto, parece existir uma semelhança entre essa espécie e o *Aedes aegypti*, a qual o classificador não é capaz de diferenciar corretamente.

Por fim, baseado nas métricas obtidas, conclui-se que é possível utilizar *Machine Learning* para realizar a classificação de mosquitos, em especial do *Aedes aegypti*, utilizando aparelhos celulares. Deve-se frisar que a quantidade de dados analisado é pequena e que possivelmente o comportamento desses mosquitos não será exatamente igual a mosquitos de outras regiões, como foi o caso das estirpes do *Anopheles gambiae* e *Anopheles arabiensis*. Portanto, essas métricas são otimistas dado que o *dataset* foi gerado em laboratório onde são geradas colônias de um ancestral comum. Em uma aplicação real, a diversidade será muito maior tornando a classificação mais desafiadora.

## 7 PROPOSTA PARA TRABALHOS FUTUROS

Nesta seção cita-se propostas para a continuação do trabalho e possíveis pontos de melhoria. No início do trabalho não se sabia se seria possível realizar a classificação de forma satisfatória, por isso, adotou-se como estratégia diferentes abordagens de classificador multiclasse, binário e *ensemble* de binários. Porém, visto que o classificador binário obteve métricas de desempenho altas, poderia ser realizado um estudo mais aprofundado da melhor arquitetura da rede CNN e seus hiperparâmetros para este modelo. Para verificar os hiperparâmetros do modelo, pode-se usar a função *GridSearchCV* da biblioteca *scikit-learn* que recebe uma grade de parâmetros e realiza o treinamento e validação diversas vezes variando os hiperparâmetros e por fim retornando apenas aqueles com as melhores métricas de desempenho.

Sugere-se como possível continuação deste projeto o desenvolvimento de um dispositivo eletrônico de baixo custo que possa ser instalado em residências e estabelecimentos comerciais para detecção em tempo real. Tal dispositivo possivelmente teria que utilizar algum artifício para atrair o mosquito o mais próximo do microfone possível. Uma alternativa ao desenvolvimento de tal dispositivo seria desenvolver um aplicativo, voltado para dispositivos móveis. Esse dispositivo precisaria estar gerando espectrogramas em tempo real e aplicando o modelo (baseado em TensorFlow) desenvolvido nesse trabalho para realizar a classificação. Em tal trabalho seria necessário aumentar a base de dados com gravações condizentes com o cenário em que o dispositivo irá atuar. Também seria necessário treinar um modelo capaz de detectar se o som gravado pertence ou não a um mosquito, caso a classificação seja positiva então classificar o mesmo utilizando as técnicas descritas nesse trabalho.

Por fim, sugere-se o uso de metadados de geolocalização e horário como *features*, visto que com esses dados poderiam ser criados diferentes modelos para diferentes regiões do mundo a fim de melhorar ainda mais o desempenho do classificador.

## REFERÊNCIAS BIBLIOGRÁFICAS

ACEVEDO, Miguel A *et al.* **Automated classification of bird and amphibian calls using machine learning: A comparison of methods**. *Ecological Informatics*, v. 4, n. 4, p. 206-214, 2009.

ALPAYDIN, Ethem. **Introduction to Machine Learning**. 3rd ed. Cambridge: The MIT Press, 2014. p. 584.

CAKIR, Emre *et al.* Convolutional recurrent neural networks for bird audio detection. In: **2017 25th European Signal Processing Conference (EUSIPCO)**, Kos, Greece: IEEE, 2017.

CHEN, Yanping *et al.* Flying Insect Detection and Classification with Inexpensive Sensors. **Journal of Visualized Experiments**, [S.l.], n. 92, 2014.

CHOLLET, François. **Deep Learning with Python**. Manning, 2017. p. 384.

DIELEMAN, Sander; SCHRAUWEN, Benjamin. End-to-end learning for music audio. In: **IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**, 39., 2014. Florence. p. 6964-6968.

FAWCETT, Tom. An introduction to ROC analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861-874, jun. 2006.

GEITGEY, Adam. **Machine Learning is Fun! Part 3: Deep Learning and Convolutional Neural Networks**. Disponível em: < <https://medium.com/@ageitgey/machine-learning-is-fun-part-3-deep-learning-and-convolutional-neural-networks-f40359318721>>. Acesso em: 10 out. 2018.

JONES, Richard. Pest Proportions. In: **Mosquito**. London: Reaktion. 2012. p. 43-56.

KAHN, M. C.; CELESTIN, W.; OFFENHAUSER, W. **Recording of Sounds produced by certain Disease-Carrying Mosquitoes**. *Science*, v. 101, n. 2622, p. 335-336, 1945.

KARN, Ujjwal. **A Quick Introduction to Neural Networks**. Disponível em: <<https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>>. Acesso em: 8 out. 2018.

KUBAT, Morislav. Performance Evaluation. In: **An Introduction to Machine Learning**. [S. l.]: Springer, 2015. cap 11, p. 216-220.

LAN, Haihan. **The Softmax Function, Neural Net Outputs as Probabilities, and Ensemble Classifiers**. Disponível em: < <https://towardsdatascience.com/the-softmax-function-neural-net-outputs-as-probabilities-and-ensemble-classifiers-9bd94d75932>>. Acesso em: 10 out. 2018.

LEUNG, Diana. **Frequency of Mosquito Wings - The Physics Factbook**. Disponível em: <<https://hypertextbook.com/facts/2000/DianaLeung.shtml>>. Acesso em: 25 out. 2018.

LI, Fei-Fei; JOHNSON, Justin; YEUNG, Serena. **CS231n Convolutional Neural Networks for Visual Recognition**. Disponível em: <<https://cs231n.github.io/neural-networks-1/>>. Acesso em: 10 out. 2018a.

LI, Fei-Fei; JOHNSON, Justin; YEUNG, Serena. **CS231n Convolutional Neural Networks for Visual Recognition**. Disponível em: <<https://cs231n.github.io/convolutional-networks/>>. Acesso em: 10 out. 2018b.

MCDONALD, Conor. **Machine learning fundamentals (II): Neural networks**. Disponível em: <<https://towardsdatascience.com/machine-learning-fundamentals-ii-neural-networks-f1e7b2cb3eef>>. Acesso em: 8 out. 2018.

MCFELL, Brian *et al.* *librosa: Audio and Music Signal Analysis in Python*. In: **Proceedings of the 14th Python in Science Conference (SciPy 2015)**, Austin, Texas: IEEE, jul, 2015.

MITCHELL, Tom. **Machine Learning**. New York: MacGraw-Hill, 1997. p. 414.

MUKUNDARAJAN, Haripriya *et al.* **Using mobile phones as acoustic sensors for high-throughput mosquito surveillance**. *eLife*, v. 6, 2017.

OFFENHAUSER, W.; KAHN, M. The Sounds of Disease-Carrying Mosquitoes. **The Journal of the Acoustical Society of America**, v. 21, n. 4, p. 462-463, 1949.

O'SHAUGHNESSY, Douglas. Hearing. In: **Speech communications: Human and Machine**. 2. ed. New York: IEEE, 2000. p. 109-139.

PACHECO, André. Redes Neurais Convolutivas – CNN. Disponível em: <<http://www.computacaointeligente.com.br/artigos/redes-neurais-convolutivas-cnn/>>. Acesso em: 10 out. 2018.

PEDREGOSA, Fabian *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825-2830, 2011.

PICZAK, Karol. Environmental Sound Classification with Convolutional Neural Networks. In: **2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)**, Boston: IEEE, nov., 2015.

REFAEILZADEH, Payam; TANG, Lei; LIU, Huan. Cross Validation. In: **Encyclopedia of Database Systems**. [S. l.], Springer, 2009. cap 4, p.532-537.

ROTHMANN, Daniel. **What's wrong with spectrograms and CNNs for audio processing?**. Disponível em: <<https://towardsdatascience.com/whats-wrong-with-spectrograms-and-cnns-for-audio-processing-311377d7ccd>>. Acesso em: 7/ 11/ 2018.

SHALEV-SHWARTZ, Shai; BEN-DAVID, Shai. **Understanding Machine Learning: From Theory to Algorithms**. [S.l.]: Cambridge University, 2014.

STEVENS, S.; VOLKMANN, J.; NEWMAN, E. A Scale for the Measurement of the Psychological Magnitude Pitch. **The Journal of the Acoustical Society of America**, v. 8, n. 3, p. 185-190, 1937.

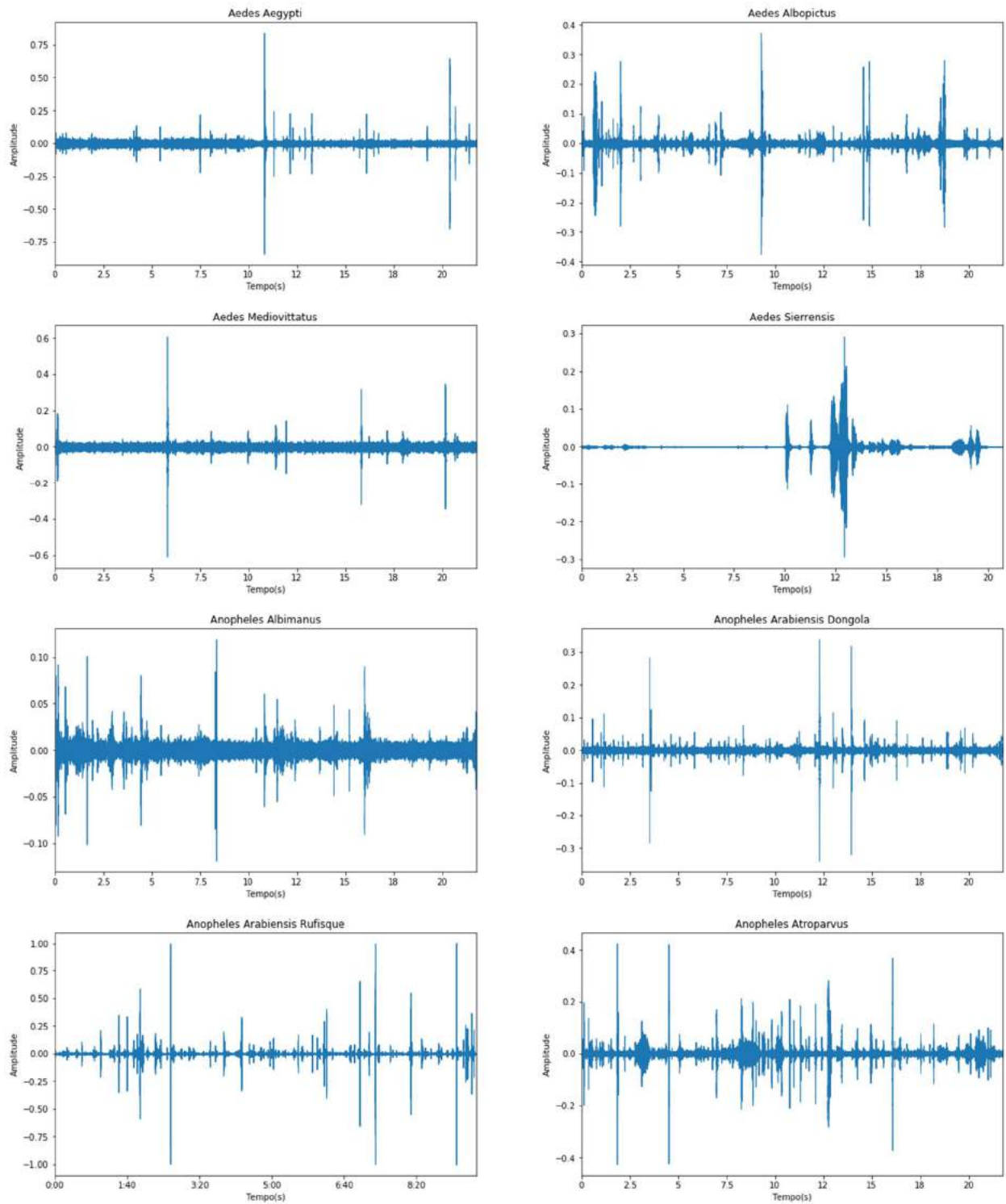
TOWNSON, H *et al.* Exploiting the potential of vector control for disease prevention. **Bulletin of the World Health Organization**, [S. l.], v. 83, n. 12, p. 942-947, Dez. 2005.

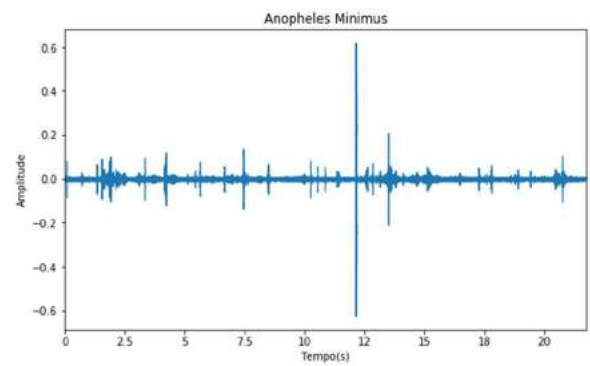
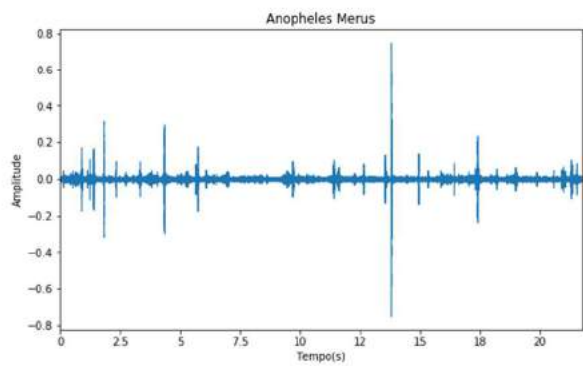
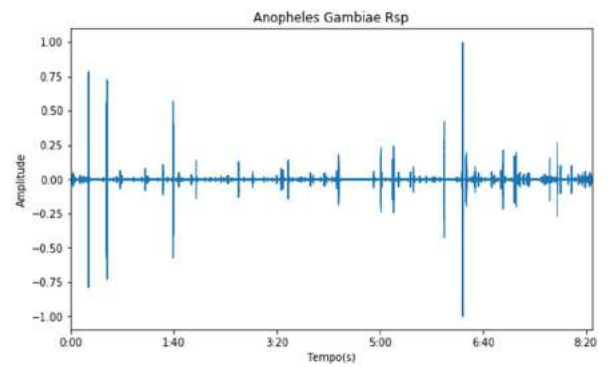
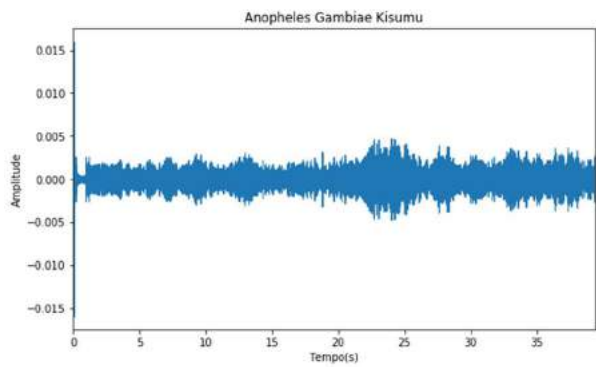
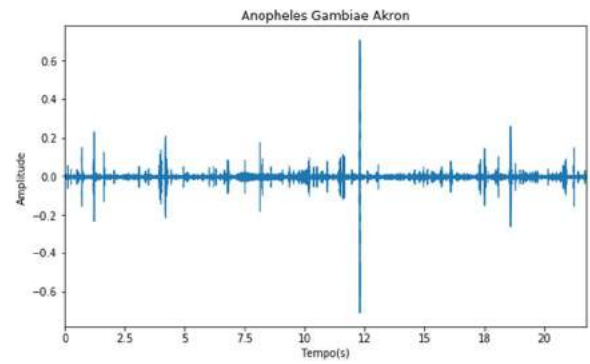
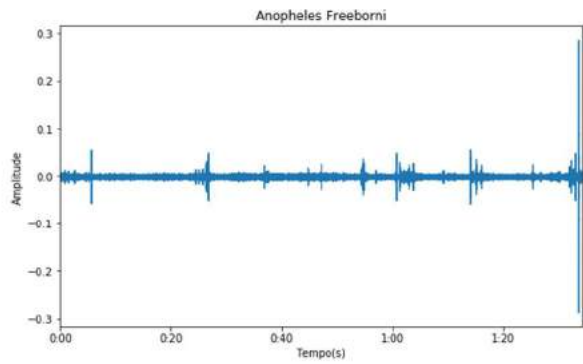
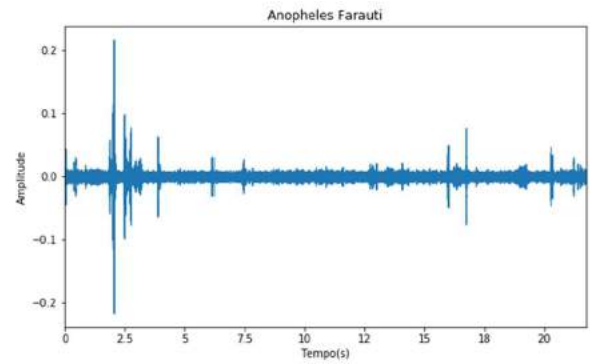
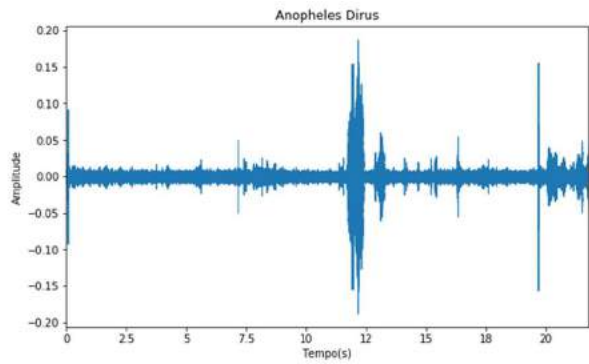
WILLIAMSON, DAVID F. The Blox Plot: A Simple Visual Method to Interpret Data. **Annals of Internal Medicine**, v. 110, n. 11, p. 916, 1989.

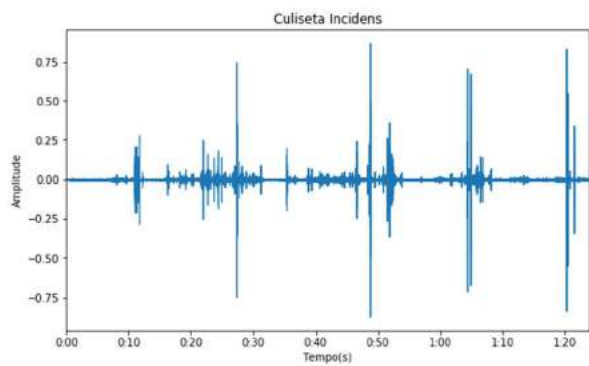
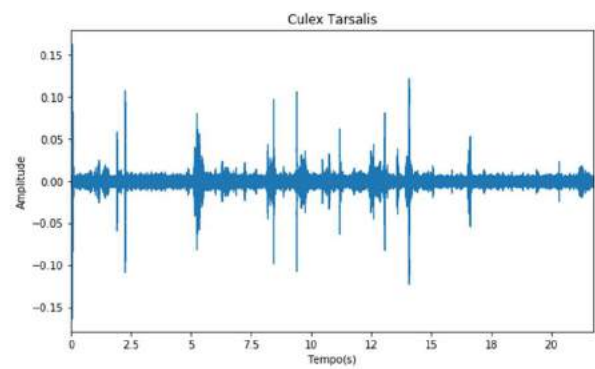
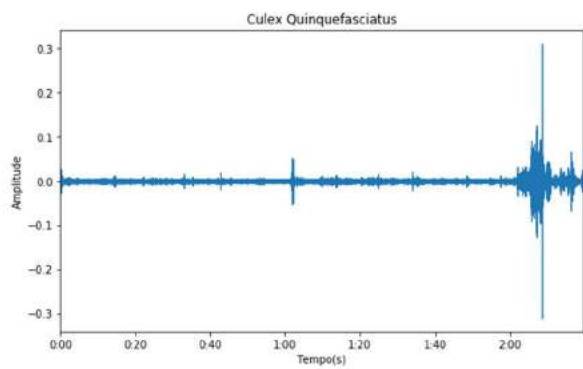
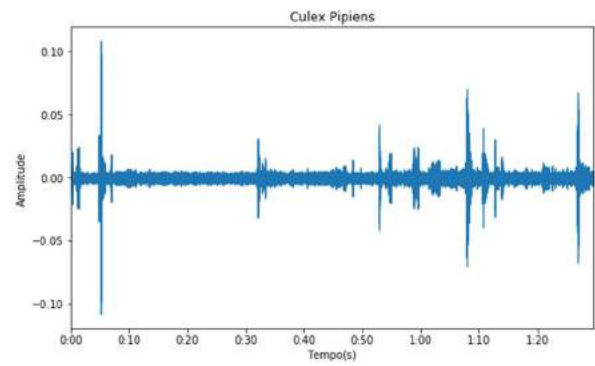
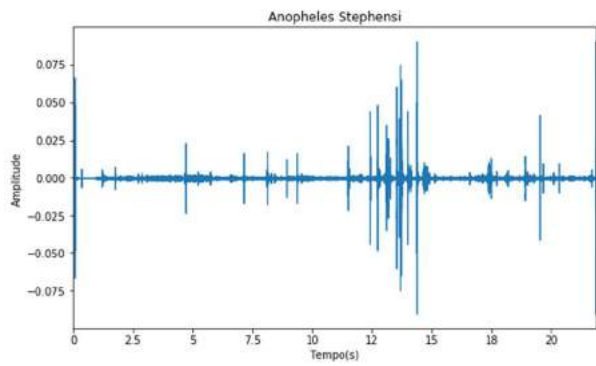
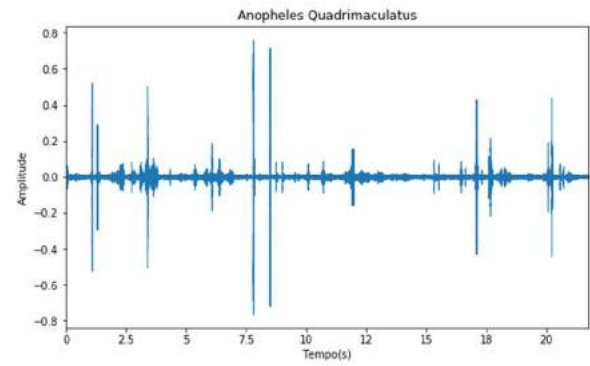
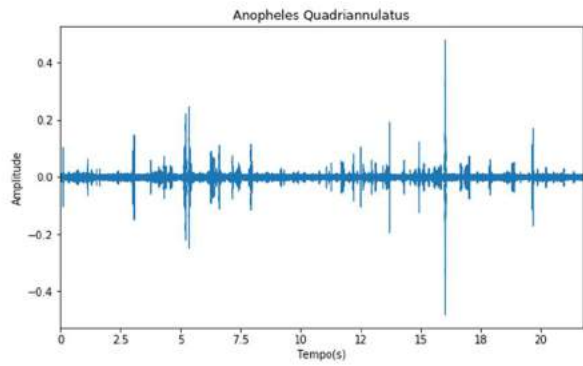
WORLD HEALTH ORGANIZATION. **A global brief on vector-borne diseases**. Geneva: World Health Organization, 2014. p. 56.

## APÊNDICE A - FORMA DE ONDA DE TODAS AS ESPÉCIES

Forma de onda de todas as espécies analisadas.



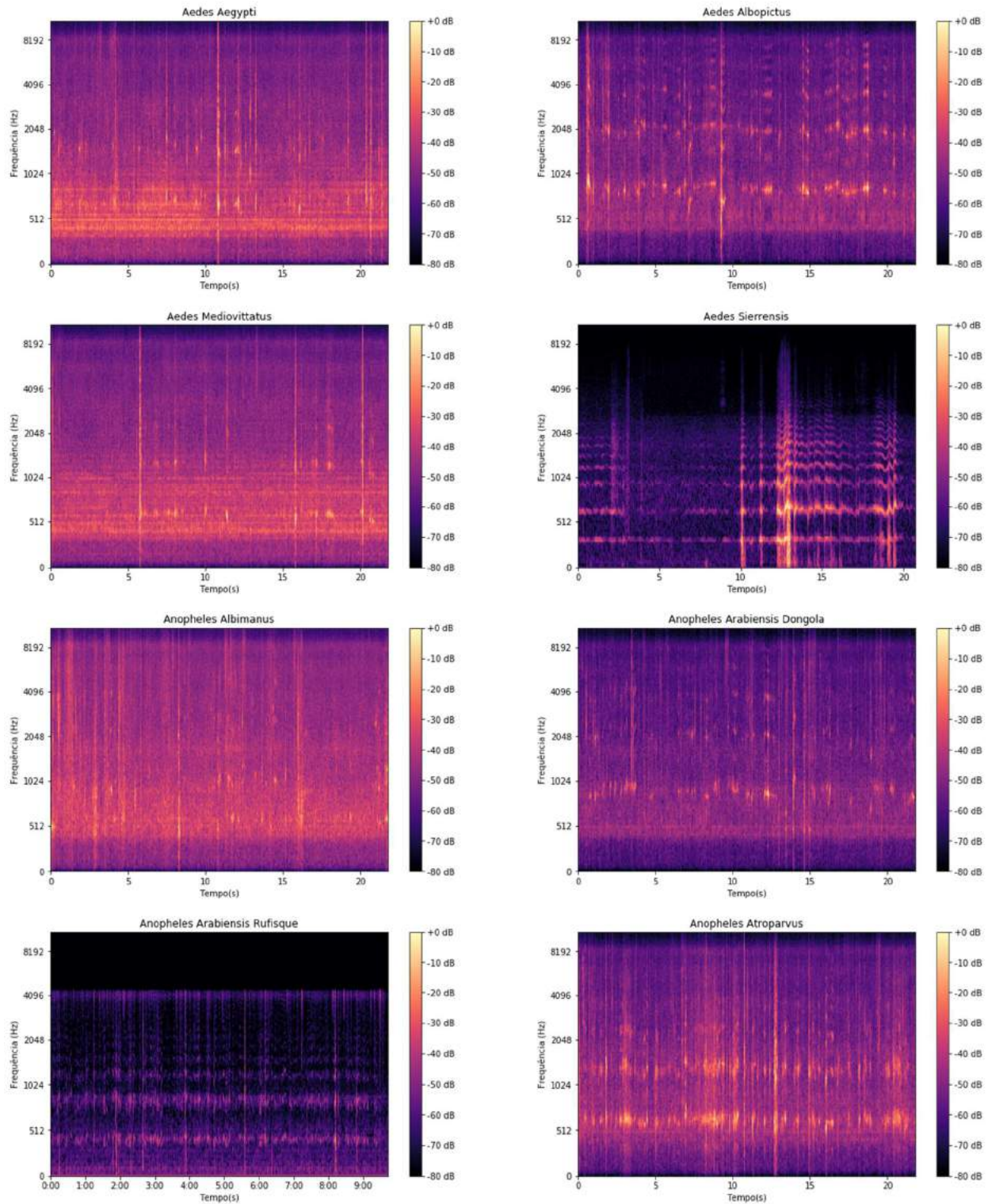


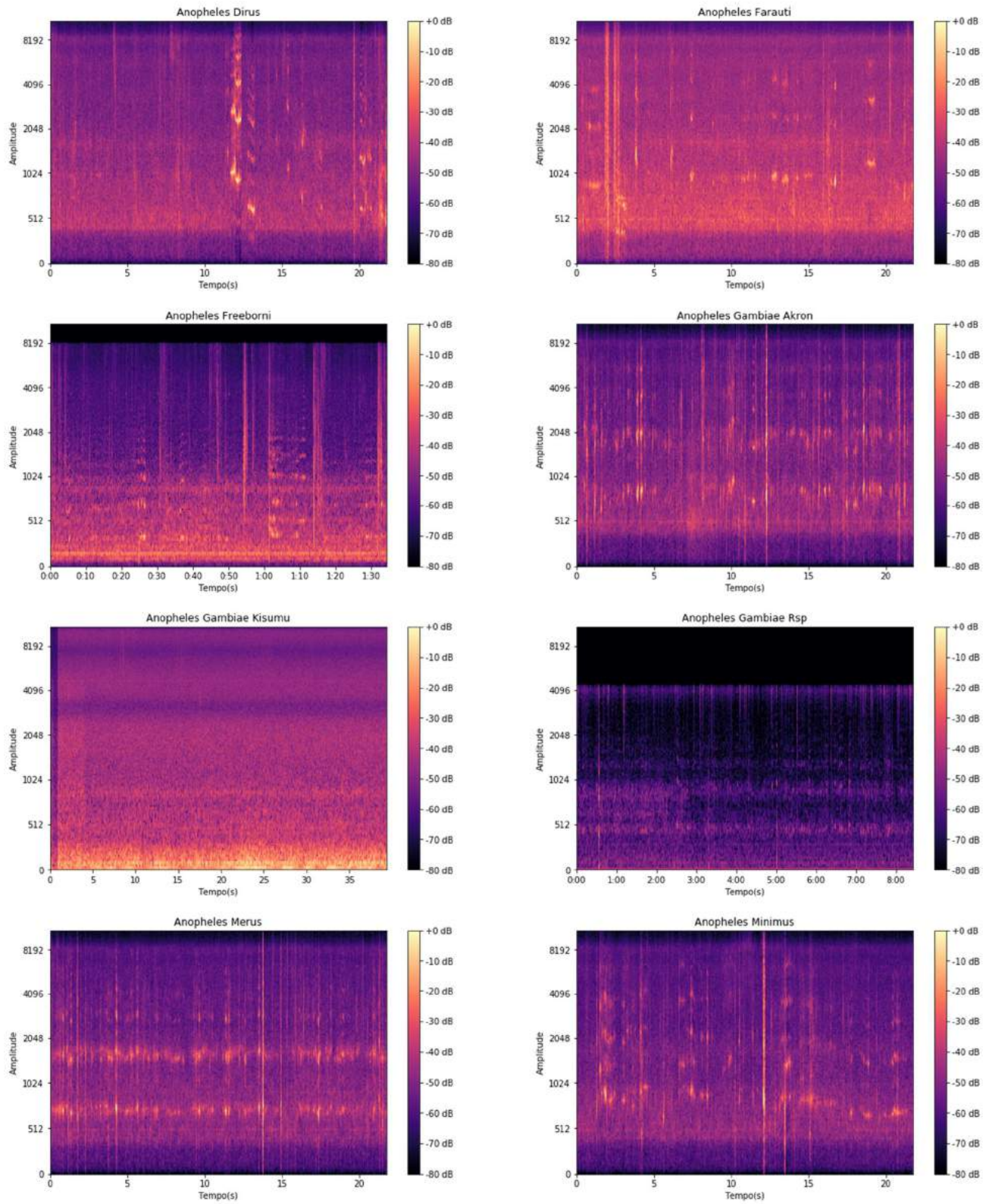




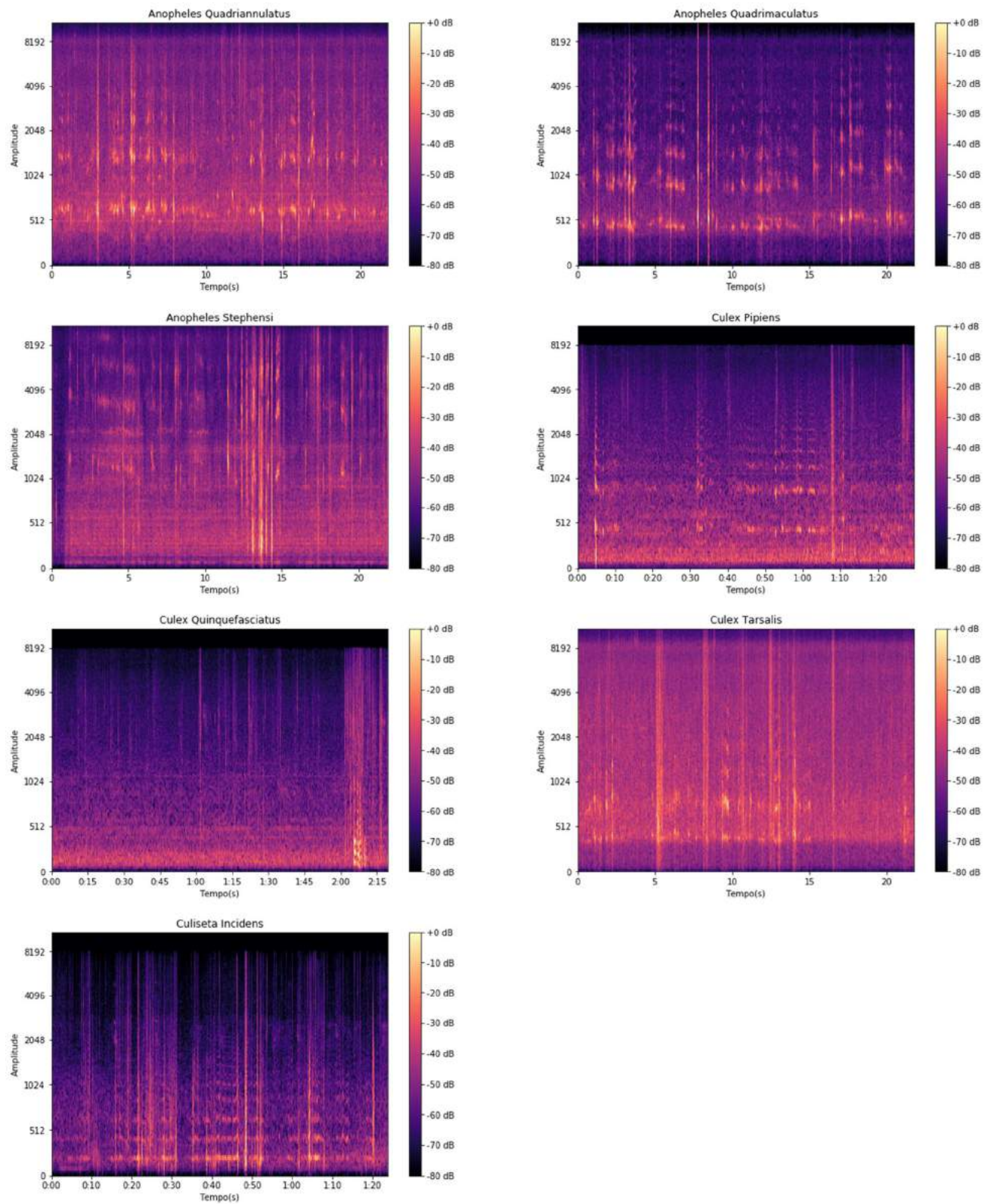
## APÊNDICE B: ESPECTROGRAMA DE TODAS AS ESPÉCIES

Espectrograma de todas as espécies analisadas, suplemento do que foi apresentado nas Figuras 16 a 20.









## APÊNDICE C: MATRIZ DE CONFUSÃO DO CLASSIFICADOR MULTICLASSE