UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA / INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM ENGENHARIA COMPUTACIONAL

Lucas Augusto Müller de Souza

Detecção de indivíduos infectados com Covid-19 através de sons de tosse forçada

Juiz de Fora

2021

Lucas Augusto Müller de Souza

Detecção de indivíduos infectados com Covid-19 através de sons de tosse forçada

Trabalho de Conclusão de Curso apresentada ao Bacharelado em Engenharia Computacional da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de Bacharel em Engenharia Computacional

Orientador: Prof. D.Sc. Heder Soares Bernardino

Coorientadores: Prof. D.Sc. Alex Borges Vieira e Jairo Francisco de Souza

Juiz de Fora

2021

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

de Souza, Lucas Augusto Müller.
Detecção de indivíduos infectados com Covid-19 através de sons de tosse forçada / Lucas Augusto Müller de Souza. – 2021.
39 f. : il.

Orientador: Heder Soares Bernardino
Coorientadores: Alex Borges Vieira e Jairo Francisco de Souza
Trabalho de Conclusão de Curso – Universidade Federal de Juiz de Fora, Faculdade de Engenharia / Instituto de Ciências Exatas. Bacharelado em Engenharia Computacional, 2021.

1. Redes Neurais Profundas. 2. Covid-19. 3. Aumento de dados em sons de tosse. I. Bernardino, Heder, orient. II. Souza, Jairo, coorient. III. Vieira, Alex, coorient. IIII. Título.

Lucas Augusto Müller de Souza

Detecção de indivíduos infectados com Covid-19 através de sons de tosse forçada

Trabalho de Conclusão de Curso apresentada ao Bacharelado em Engenharia Computacional da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de Bacharel em Engenharia Computacional

Aprovada em 9 de Setembro de 2021

BANCA EXAMINADORA

Prof. D.Sc. Heder Soares Bernardino - Orientador
Universidade Federal de Juiz de Fora

Prof. D.Sc. Jairo Francisco de Souza - Coorientador
Universidade Federal de Juiz de Fora

Prof. D.Sc. Alex Borges Vieira - Coorientador
Universidade Federal de Juiz de Fora

Prof. D.Sc. Saulo Moraes Villela
Universidade Federal de Juiz de Fora

Prof. D.Sc. Leonardo Goliatt da Fonseca
Universidade Federal de Juiz de Fora

Dedico este trabalho ao meu pai, minha mãe e minha irmã, que me apoiaram e me incentivaram durante toda minha trajetória.

AGRADECIMENTOS

Agradeço aos meus pais, Francisco e Adriana, e minha irmã Júlia, por me apoiarem e sempre me incentivarem a buscar novos conhecimentos. Espero um dia conseguir recompensar vocês por tudo que fizeram por mim.

A todos os professores e funcionários que me ajudaram nessa caminhada, desde a base no Colégio de Aplicação João XXIII, até a graduação na Universidade Federal de Juiz de Fora. Gostaria de agradecer especialmente meus orientadores Heder Soares Bernardino, Jairo Francisco de Souza e Alex Borges Vieira, por toda a paciência e conhecimentos passados. Além disso, gostaria de agradecer também ao meu tutor em educação tutorial, Elson Magalhães Toledo. Participar do Grupo de Educação Tutorial da Engenharia Computacional foi uma das experiências mais enriquecedoras que eu vivi.

Aos meus amigos que estiveram comigo durante toda essa jornada.

A Universidade Federal de Juiz de Fora, por proporcionar um ensino público de qualidade, e às agências de fomento.

RESUMO

O novo coronavírus (COVID-19) é uma doença infecciosa declarada uma pandemia em 2020 pela Organização Mundial da Saúde (OMS). Através de muita cooperação e do esforço de cientistas ao redor do mundo, diversas vacinas foram criadas. Entretanto, mesmo que grande parte da população mundial seja vacinada, não existem garantias de que o vírus irá desaparecer. Portanto, métodos com custo baixo, não-invasivos e capazes de gerar resultados em tempo-real são importantes para detectar indivíduos infectados e possibilitar um tratamento adequado mais rapidamente, além de evitar o espalhamento do vírus. Na literatura é possível encontrar modelos computacionais capazes de distinguir uma pessoa saudável de uma pessoa infectada pela COVID-19, utilizando conjuntos de dados de tosse forçada coletados de indivíduos ao redor do mundo. Um grande desafio existente está no desbalanceamento desses dados, tendo em vista que existem mais amostras de indivíduos saudáveis do que de contaminados. Neste trabalho, são propostas alterações em um modelo de Redes Neurais Profundas, disponível na literatura, assim como no treinamento do mesmo. Além disso, foram realizados estudos do aumento de dados nesses áudios. Os resultados mostram que o modelo adaptado além de ser mais simples (possuir menos parâmetros) consegue generalizar melhor a predição de infectados, apresentando uma Área Sob a Curva ROC média de 0,885 e intervalo de confiança (0,881 - 0,888), contra 0,771 e (0,752 - 0,783) do modelo original.

Palavras-chave: Redes Neurais Profundas. Covid-19. Aumento de dados. Sons de tosse.

ABSTRACT

The new coronavirus (COVID-19) is an infectious disease declared a pandemic in 2020 by the World Health Organization (WHO). Through a lot of cooperation and the effort of scientists around the world, several vaccines were created. However, even if a large part of the world's population is vaccinated, there is no guarantee that the virus will ever disappear. Therefore, non-invasive methods, with low cost, and capable of generating results in real-time are important to detect infected individuals and enable an adequate treatment quicker, in addition to preventing the spread of the virus. In the literature it is possible to find computational methods capable of distinguishing a healthy person from an infected with COVID-19 with high accuracy, using data sets of forced cough, collected online from individuals around the world. A great existing challenge is in the unbalance of these data, considering that there are more samples of healthy individuals than contaminated ones. In this work, we proposed changes in a Deep Neural Networks model, available in the literature, as well as in its training. In addition, we carried out studies on the data augmentation of these audios. The results show that the adapted model is simpler (having fewer parameters) and manages to better generalize the prediction of infected individuals, presenting an average Area Under the Curve ROC (AUC) of 0.885 and a confidence interval (0.881 - 0.888), against 0.771 and (0.752 - 0.783) of the original one.

Keywords: Deep Neural Networks. Covid-19. Data augmentation. Cough sounds.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

OMS	Organização Mundial da Saúde
IA	Inteligência Artificial
AUC	*Area Under the ROC Curve*
ML	*Machine Learning*
MFCC	*Mel Frequency Cepstral Coefficients*
TDC	Transformada Discreta do Cosseno
XGB	*eXtreme Gradient Boosting*
DNN	*Deep Neural Network*
CNN	*Convolutional Neural Network*
NN	*Neural Network*
FC	*Fully Connected*
ReLU	*Rectified Linear Unit*
ES	*Early Stopping*
Sens.	Sensibilidade
Esp.	Especificidade
VPP	Valor Preditivo Positivo
VPN	Valor Preditivo Negativo

SUMÁRIO

1 INTRODUÇÃO

Em 11 de março de 2020, a Organização Mundial da Saúde (OMS) declarou a pandemia de Covid-19, causada pelo novo coronavírus (Sars-Cov-2), devido à sua rápida disseminação geográfica. Até 15 de agosto de 2021, já foram registrados no total mais de 205 milhões de casos confirmados e mais de 4,3 milhões de mortes em decorrência da doença[1]. Nesse mesmo período, foram confirmados mais de 20,5 milhões de casos e mais de 574 mil mortes no Brasil. Após muito esforço de cientistas ao redor do mundo, em dezembro de 2020, foi iniciado o processo de vacinação da população ao redor do mundo. Atualmente, mais de 4,68 bilhões de doses foram ministradas em 183 países[2]. No Brasil, foram aplicadas mais de 165 milhões de vacinas contra a Covid-19. Apesar disso, mesmo que a maior parte dos indivíduos seja vacinada, não há garantias que o vírus irá desaparecer completamente. Portanto, a melhor forma de evitar que o vírus se espalhe e garantir que pessoas infectadas pela Covid-19 tenham acesso ao tratamento adequado o mais breve possível, é através da ampla testagem e do isolamento daqueles indivíduos contaminados. Neste momento, a melhor forma de testagem é através do teste *Reverse Transcription Polymerase Chain Reaction* (RT-PCR). Entretanto, o resultado deste teste pode levar dias para ser obtido, além de não ser barato e de haver uma limitação na quantidade de testes que podem ser feitos diariamente, a depender da infra-estrutura e da quantidade de matéria-prima disponível para realização do mesmo.

Na literatura existem diversos estudos de detecção de doenças utilizando gravações de sons humanos, tais como fala, respiração e tosse, através de técnicas de Inteligência Artificial (IA). Nestes estudos são utilizadas diversas técnicas de análise acústica, capaz de extrair informações desses áudios, que serão utilizadas no treinamento de modelos de classificação para discriminação de indivíduos que possuam a doença daqueles que não a possuem. A seguir, serão apresentados estudos que utilizaram algoritmos de Aprendizado de Máquina (*Machine Learning* - ML) na detecção de doenças como: Parkinson, Esclerose Lateral Amiotrófica e tuberculose.

Braga et al. (2019) e seus associados treinaram diferentes modelos de AM na detecção da doença de Parkinson baseado na análise de gravações de fala de pacientes. Por se tratar de um problema de classificação binária, com a classe de cada amostra previamente conhecida, os modelos utilizados nessa tarefa foram: Floresta Aleatória, Redes Neurais Artificiais (*Neural Network* - NN) e Máquina de Vetores de Suporte. Os dados utilizados consistem de três bases de dados com gravações de fala. A primeira é composta por 22 pessoas que possuem a doença de Parkinson, totalizando 88.8 minutos de falas gravadas. Já a segunda, é formada por 30 pessoas saudáveis, totalizando 28.2 minutos de áudio. Além disso, foi utilizada uma terceira base de dados disponível na

[1] `https://covid19.who.int/`

[2] `https://www.bloomberg.com/graphics/covid-vaccine-tracker-global-distribution`

literatura para validar o desempenho dos modelos utilizados. Os resultados reportados demonstram que a Floresta Aleatória foi o modelo com melhor desempenho conforme as métricas utilizadas, e obteve uma acurácia de 99,94%.

Além disso, Vashkevich (Vashkevich et al., 2019) treinou um classificador K-vizinhos mais próximos na detecção de disfunção bulbar em pacientes com Esclerose Lateral Amiotrófica (ELA), que se trata de uma doença do sistema nervoso que causa a morte dos neurônios responsáveis pelo controle dos músculos voluntários. A base de dados utilizada é formada por gravações de 54 pessoas, sendo 39 falantes saudáveis e 15 pacientes com ELA. Todos os participantes produziram a vogal sustentada /*a*/ em um tom confortável e sonoridade constante pelo maior tempo possível. Esta fonação foi executada em uma respiração. Características acústicas extraídas a partir dessas gravações foram utilizadas no treinamento do modelo. Os autores reportam que o melhor resultado obtido pelo modelo possui uma acurácia de 90,7%, com uma sensibilidade de 86,7% e uma especificidade de 92,2%. A sensibilidade corresponde ao percentual de resultados positivos obtido pelo modelo dentre as pessoas com uma determinada doença, enquanto a especificidade é a proporção de resultados negativos do método nos indivíduos que não apresentam a doença.

Ainda, Botha et al. (2018) utilizou sons de tosse forçada e informações médicas para detecção de pacientes infectados com tuberculose, que ainda se trata de uma das doenças mais mortais no mundo. A base de dados utilizada era composta por gravações de tosse de 38 indivíduos, sendo 17 pacientes infectados com a doença e 21 saudáveis. Além disso, foram adicionadas 5 informações clínicas: (i) circunferência do braço; (ii) temperatura; (iii) índice de massa corporal; (iv) presença de conjuntiva pálida e (v) frequência cardíaca. Um modelo de Regressão Logística foi treinado utilizando esses dados, e obteve no melhor caso uma Área Sob a Curva ROC (*Area Under the ROC Curve* - AUC) de 0,95 com uma acurácia de 78%.

Estes estudos demonstram a capacidade que algoritmos de IA possuem na detecção de diferentes doenças, baseado em análises acústicas de sons produzidos por seres humanos. Na literatura existem trabalhos promissores na utilização de métodos de IA na detecção de pacientes infectados com a Covid-19, utilizando gravações de tosse forçada. Estes trabalhos serão apresentados na Seção 2.3. Considerando que esses modelos possuem a capacidade de classificação de pacientes infectados, eles podem ser adotados em larga escala, dado que seria possível a utilização de aparelhos celulares e computadores para a gravação de tosses forçadas que seriam alimentadas ao modelo. Dessa forma, o objetivo deste trabalho foi propor um modelo capaz de diferenciar indivíduos infectados pela Covid-19, que possa ser usados para detecção da doença, com baixo custo, de forma não-invasiva, e com a capacidade de geração de resultados em tempo real. Vale destacar que esses métodos não devem ser utilizados como fator determinante para o diagnóstico de um paciente, mas sim como uma forma de pré-triagem para a realização de testes médicos, tais como o teste

RT-PCR.

Neste trabalho, foram propostas modificações na estrutura de um modelo de Redes Neurais Profundas (*Deep Neural Network* - DNN) (Chaudhari et al., 2020) para a detecção de pacientes infectados com Covid-19 através da análise de características acústicas de sons de tosse forçada e dos sintomas reportados pelos indivíduos. Além disso, foi realizado um estudo sobre a aplicação de aumento de dados nas gravações de tosse, pois as bases de dados apresentam um grande desbalanceamento, contendo poucas amostras na classe positiva (pacientes infectados com a doença). Os resultados obtidos foram comparados com o modelo utilizado de referência na literatura e comprovam que as modificações propostas aumentaram a capacidade de classificação do modelo. As contribuições desse trabalho são um modelo de DNN mais simples que o disponível na literatura, capaz de obter melhores resultados na classificação de pessoas infectadas com a Covid-19. Além de um método de aumento de dados aplicado a gravações de tosse.

O restante do trabalho está organizado da seguinte forma. No Capítulo 2 serão apresentados conceitos básicos relacionados a onda sonora digital, como ela pode ser utilizada no treinamento de modelos e os trabalhos relacionados presentes na literatura. Os materiais e métodos utilizados serão descritos no Capítulo 3. Já no Capítulo 4, serão descritos os experimentos computacionais realizados, as métricas utilizadas e os resultados alcançados. No Capítulo 5 serão apresentadas as conclusões do trabalho, suas limitações, e possíveis caminhos para trabalhos futuros. Finalmente, no Apêndice A foi discutido o método proposto de aumento de dados em gravações de áudios.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 A ONDA SONORA DIGITAL

Para se entender as aplicações de algoritmos de IA baseados em ondas sonoras, é preciso entender primeiro como esta é representada digitalmente. Ondas sonoras são formadas por complexas sequências de compressões e rarefações do ar (Jurafsky and Marin, 2008), que no caso de sons humanos, tais como a fala, respiração, e tosse são causadas pela forma com que o ar passa pelas cordas vocais e sai pelas cavidades, nasal ou oral.

O processo de conversão de uma onda sonora para seu formato digital se inicia com a captura da mesma através de um microfone, que a converte em uma corrente elétrica, e posteriormente em um sinal digital. Essa representação digital é formada por dois processos, chamados *sampling* e *quantization*. O primeiro processo denominado *sampling* consiste na medição da amplitude da onda em um determinado momento (Jurafsky and Marin, 2008), e o número de medições por segundo é chamado *sampling rate*. A Figura 1 possui um exemplo de uma onda senoidal com 10 *samples* por segundo. Para representarmos uma onda sonora digital corretamente é necessário ao menos 2 *samples* por ciclo, um responsável por medir a amplitude positiva da onda e o outro a amplitude negativa. Quanto mais medições por segundo forem coletadas de uma determinada onda sonora mais fielmente ela pode ser representada. Um *sampling rate* de 16kHz é muito utilizado para microfones atualmente (Jurafsky and Marin, 2008), o que consiste na obtenção de 16.000 *samples* por segundo de áudio. Usualmente, cada *sample* é armazenado digitalmente como um inteiro de 8 bits (valores de -128 a +127) ou 16 bits (valores de -32.768 a +32.767). Esse processo de representação de um valor real (a amplitude da onda) por um valor inteiro é chamado *quantization*. Após os processos de *sampling* e *quantization*, a onda sonora digital pode ser armazenada em diversos formatos.

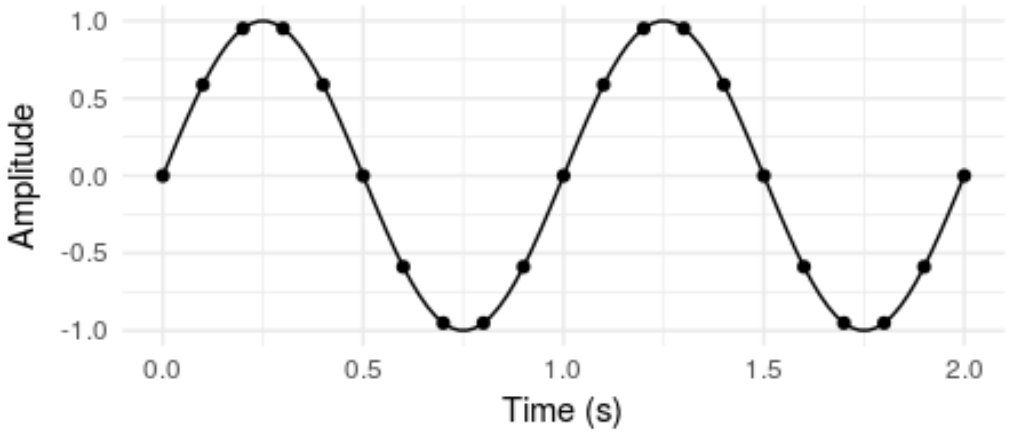


Figura 1 – Representação de uma onda senoidal com 10 *samples* por segundo. Fonte: Camero et al. (2018)

.

2.2 EXTRAÇÃO DE INFORMAÇÕES DE UMA ONDA SONORA

Uma importante propriedade de um som é o tom, que se trata de uma sensação mental da frequência fundamental, conforme explicado por (Jurafsky and Marin, 2008). A frequência fundamental ou F0 é a frequência de vibração das cordas vocais. A percepção de tom pelo ouvido humano é mais precisa entre 100 Hz e 1000 Hz, que varia linearmente com a frequência fundamental. Entretanto, para frequências maiores que 1000 Hz, o tom varia logaritmicamente com a frequência. Portanto, as diferenças entre frequências maiores não são percebidas com tanta precisão.

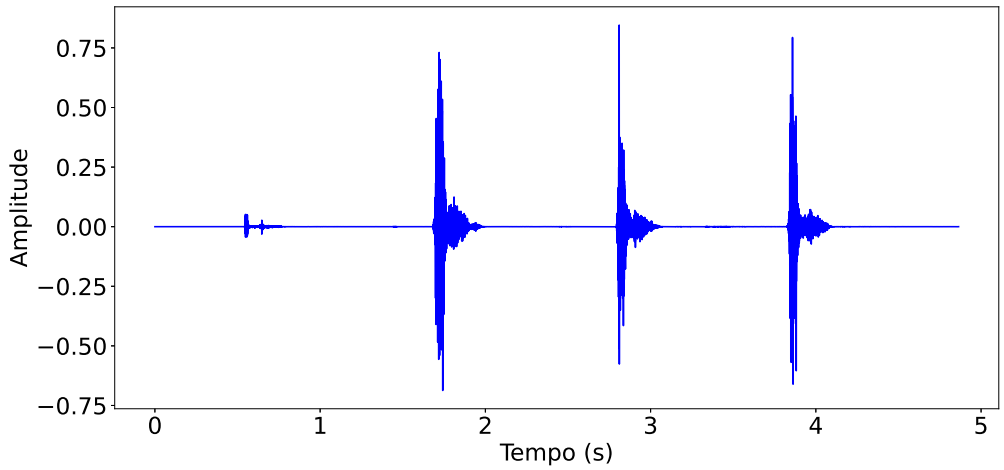
Existem diversos modelos psicoacústicos que consideram a percepção não linear do som pelo ouvido humano. Uma das escalas mais utilizadas, denominada escala mel, foi proposta por Stevens and Volkmann (1940). Esta escala explora a relação de percepção da frequência fundamental entre dois tons, de forma que pares de sons que são perceptualmente equidistantes no tom sejam separados pela mesma quantidade de mels, conforme explicado em Jurafsky and Marin (2008) e Fachini and Heinen (2016). A conversão da frequência original (f_o) para a frequência na escala mel (f_{mel}) pode ser calculada da seguinte forma:

$$f_{mel}(f_o) = 1127 \ln\left(1 + \frac{f_o}{700}\right) \tag{2.1}$$

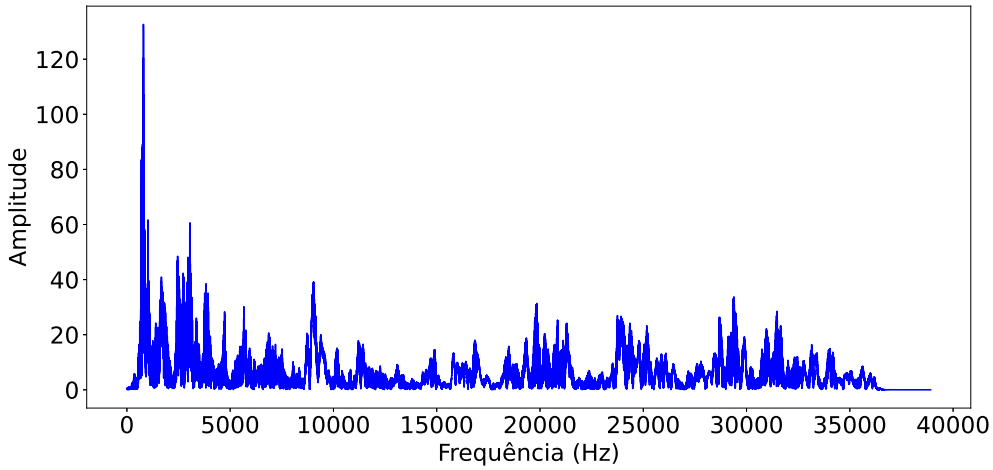
Geralmente, algoritmos de ML treinados com ondas sonoras não utilizam o áudio como entrada do modelo, mas sim características extraídas dos sons. Existem diversas informações que podem ser extraídas e utilizadas como entrada dos modelos, entretanto neste trabalho nos concentramos nas características utilizadas no treinamento do modelo implementado.

Uma informação muito utilizada é o Espectrograma Mel. Este baseia-se no Teorema de Fourier que diz que qualquer função de onda complexa pode ser aproximada pelo somatório de ondas senoidais e cossenoidais de diferentes frequências. Aplicando a transformada de Fourier sobre uma onda sonora é possível decompor seu sinal nas diferentes frequências que formam o som, bem como as amplitudes de cada uma das frequências, ou seja, o sinal é convertido de um domínio temporal para o domínio da frequência. O resultado dessa conversão é chamado espectro. A Figura 2 possui o exemplo de uma gravação de tosse e o espectro correspondente.

Um espectro consegue mostrar as frequências que formam uma onda e suas respectivas amplitudes em um determinado instante, entretanto, no caso da tosse essas frequências mudam temporalmente. Portanto, pode-se utilizar um espectrograma para entender como as diferentes frequências que formam uma onda sonora mudam ao longo do tempo. A Figura 3 mostra o espectrograma correspondente ao som da tosse mostrado anteriormente. No eixo-x da imagem pode-se observar o tempo em segundos, no eixo-y a frequência da onda e a cor de cada ponto em um espectrograma corresponde à amplitude da frequência.



(a) Exemplo de uma gravação de tosse forçada, retirada da base de dados utilizada no treinamento do modelo.



(b) Espectro correspondente ao exemplo de gravação de tosse forçada.

Figura 2 – Exemplo de uma gravação de tosse forçada e seu espectro correspondente.

Outra forma de extração de informação de ondas sonoras bastante utilizado na literatura para detecção de doenças são os *Mel Frequency Cepstral Coefficients* (MFCC) (Laguarta et al., 2020; Sonu and Sharma, 2012; Chaudhari et al., 2020; Brown et al., 2020). Os MFCC imitam a percepção não-linear do som pelo ouvido humano, e possuem uma grande capacidade de síntese, podendo reduzir um espectro de 1024 pontos para cerca de 15 a 40 pontos, conforme citado por (Fachini and Heinen, 2016).

O processo de extração dos MFCCs se inicia com a divisão do sinal sonoro em janelas, seguido da aplicação da Transformada Rápida de Fourier em cada uma das janelas para converter o sinal de entrada do domínio temporal para o domínio da frequência. Em seguida, esse sinal passará um banco de filtros passa-banda com envelopes triangulares, linearmente espaçados até 1000 Hz e logaritmicamente espaçados a partir de dessa frequência (assim como a escala mel, e a percepção das ondas sonoras pelo ouvido humano). Em seguida, será aplicada a função logarítmica ao sinal de saída do banco de filtros. A última etapa consiste

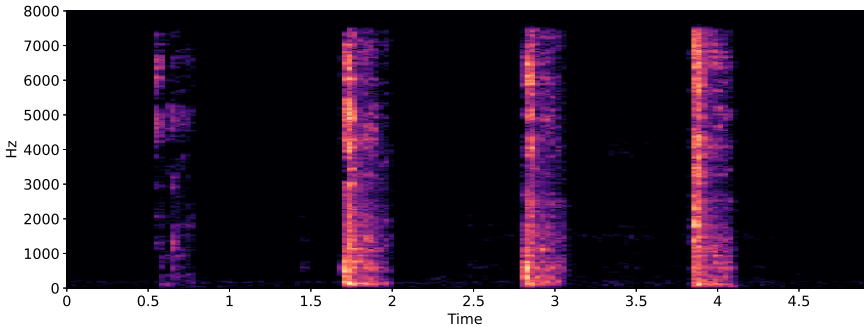


Figura 3 – Exemplo de um espectrograma calculado a partir de uma tosse forçada.

na aplicação da Transformada Discreta do Cosseno (TDC), para converter o Espectro Log Mel para o domínio temporal. As informações contidas no zerésimo coeficiente são frequentemente substituídas ou aumentadas pela energia logarítmica. O resultado dessa conversão é chamado *Mel Frequency Cepstrum Coefficients*. Esse processo é mostrado na Figura 4.

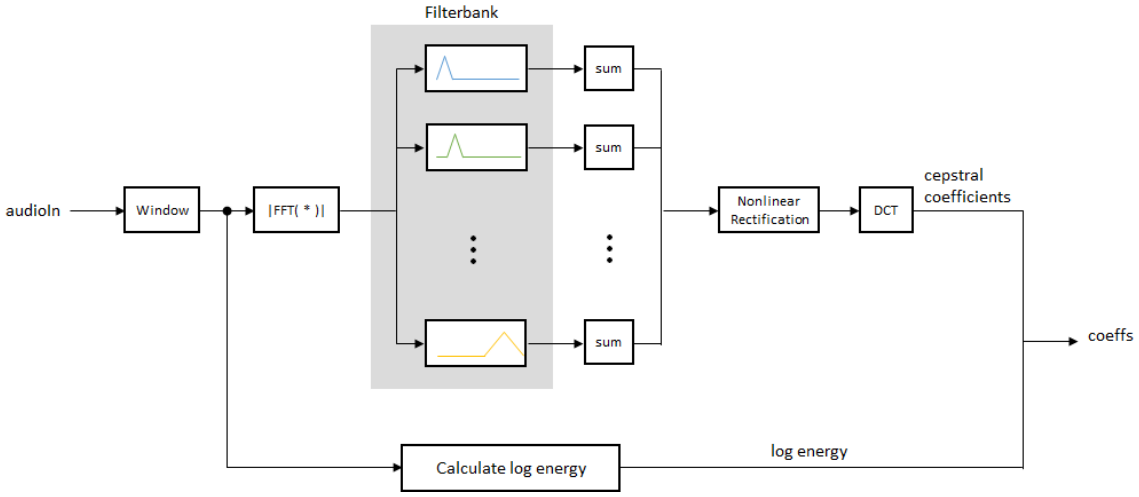


Figura 4 – Representação do processo de extração dos *Mel Frequency Cepstral Coefficients*, imagem retirada de `https://www.mathworks.com/help/audio/ref/mfcc.html`.

2.3 TRABALHOS RELACIONADOS

Desde que a Covid-19 foi declarada uma pandemia em 2020, muitos grupos (Brown et al., 2020; Laguarta et al., 2020; Orlandic et al., 2021; Sharma et al., 2020) começaram a coletar dados no formato *crowdsourcing* através de aplicativos de celular e *sites* na *internet*, com o objetivo de montar sistemas capazes de detectar indivíduos contaminados com a Covid-19. Estes algoritmos podem ser utilizados como pré-triagem para a realização de exames médicos, tais como o teste RT-PCR, através da identificação de indivíduos com maior probabilidade de estarem infectados. Outra possibilidade de aplicação dos algoritmos, é para o rastreamento de contato e quarentena preventiva para indivíduos que

tiveram contato recente com um paciente com alta probabilidade de infecção, até que o exame médico seja realizado. Os dados coletados por estes grupos englobam sons humanos (tosse, respiração e/ou fala), informações médicas, sintomas e diagnóstico de pacientes.

A base de dados COUGHVID (Orlandic et al., 2021) contém mais de 20.000 gravações de tosses, coletadas entre 1 de abril de 2020 e 10 de setembro de 2020, através de um *site* implantado em um servidor privado. Segundo os autores, esta base possui indivíduos com uma ampla gama de idades, gêneros, localizações geográficas, condições respiratórias pré-existentes e estado de saúde (infectado ou saudável), com o potencial de permitir que modelos computacionais consigam obter uma boa generalização. Apesar da abundância de gravações disponíveis, apenas 1.010 destas amostras foram gravadas por pacientes que alegam estarem infectados com a Covid-19. Uma desvantagem observada nas bases de dados coletadas *online* está no fato de que o status de Covid-19 de um indivíduo (infectado ou saudável), pode ser um auto-diagnóstico, sem necessariamente ter realizado algum exame médico, dessa forma, sujando os dados e potencialmente confundindo o modelo. Para tentar minimizar este problema e legitimar que amostras marcadas como Covid-19 positivo tenham realmente vindo de indivíduos infectados, Orlandinc e seus associados (Orlandic et al., 2021) analisaram a localização geográfica de onde as amostras vieram. Os autores combinaram estatísticas de novos casos da OMS[1], com a base de populações de 2019 das Nações Unidas[2], para determinar a taxa de infecção no país de origem da amostra 14 dias antes dela ser fornecida. Segundo os autores, essa análise revelou que 94.4% das gravações de pacientes infectados se originaram de países com mais de 20 casos confirmados por milhão de habitantes.

Para evitar amostras que não contenham nenhuma tosse, os autores do artigo treinaram um classificador eXtreme Gradient Boosting (XGB), com base em 121 sons de tosse e 94 não-tosses, com o objetivo de determinar a probabilidade de uma determinada gravação conter um som de tosse. Este modelo obteve uma AUC de $0,97$, tendo sido utilizado para classificar os áudios disponíveis na base de dados. As probabilidades de saída do classificador foram incluídas nos metadados de cada registro sob o rótulo *cough_detected*.

O objetivo do projeto Coswara (Sharma et al., 2020) segundo os autores é criar uma base de dados com amostras sonoras de indivíduos saudáveis e contaminados. Diferente da base COUGHVID, no projeto Coswara foram coletados 9 categorias diferentes de sons, sendo elas: respiração (breve e profunda), tosse (curta e pesada), sustentação vocálica de três tipos diferentes, e contagem de dígitos do 1 ao 20 (velocidades normal e rápida). Além disso, foram coletados metadados com informações sobre o gênero, idade, localização, estado de saúde (saudável, exposto, curado ou infectado) e a presença de condições médicas pré-existentes. Assim como na base de dados COUGHVID a coleta dos dados aconteceu *online*, através de um *site* que pode ser acessado pelo computador ou pelo celular, onde o

[1] `https://data.humdata.org/dataset/coronavirus-covid-19-cases-and-deaths`
[2] `https://population.un.org/wpp/Download/Standard/CSV/`

indivíduo deve fornecer os metadados citados anteriormente, bem como as 9 categorias de gravações. Entretanto, neste projeto, o processo de limpeza e verificação da qualidade das amostras é feito manualmente, e ainda está ocorrendo. Na primeira versão, publicada em 7 de agosto de 2020 em um repositório do GitHub[3] o projeto possuía dados de 941 participantes. No momento do desenvolvimento deste trabalho o projeto contém 1575 amostras, onde há 109 registros de pacientes infectados com Covid-19 rotulados como *positive_asymp*, *positive_mild*, ou *positive_moderate*, e 1476 registros de pacientes com Covid-19 negativo rotulados como *healthy*, *no_resp_illness_exposed*, *recovered_full*, ou *resp_illness_not_identifier*.

Na literatura existem estudos que mostram resultados promissores na detecção de pacientes infectados com Covid-19 baseado em sons humanos. Por exemplo, destacam-se as contribuições de um grupo de pesquisadores do MIT (Laguarta et al., 2020), de pesquisadores do grupo Virufy (Chaudhari et al., 2020) e de Brown (Brown et al., 2020) e seus associados. Todos esses trabalhos utilizaram sons de tosse forçada no treinamento do modelo, entretanto, Chaudhari et al. (2020) adicionaram informações médicas de sintomas informados pelos indivíduos além da tosse, enquanto Brown et al. (2020) utilizaram também sons de respiração. Em relação à extração de características dos sinais sonoros, o estudo de Laguarta et al. (2020) utilizou o MFCC para treinamento do modelo. Enquanto, o trabalho de Chaudhari et al. (2020) além do MFCC utilizou também o Espectrograma Log Mel. Já, Brown et al. (2020) extraíram diversas características do sinal, alguns exemplos são: *Duration, Onset, Tempo, Period, RMS Energy, Spectral Centroid, entre outros.* No total foram extraídas 477 características dos sons de tosse e respiração. No que diz respeito às bases de dados utilizadas no treinamento do modelo, todos os trabalhos utilizaram bases distintas. Brown et al. (2020) e Laguarta et al. (2020) utilizaram bases de dados próprias, coletadas online, enquanto a primeira se encontra disponível através da assinatura de um acordo para utilização dos dados, a segunda não foi disponibilizada. Por outro lado, Chaudhari et al. (2020) combinaram dados das bases COUGHVID e Coswara, que se encontram disponíveis para a utilização de toda a comunidade científica.

Os estudos de Laguarta et al. (2020) e Chaudhari et al. (2020) utilizaram modelos de DNN, enquanto Brown et al. (2020) utilizaram modelos de ML. Laguarta et al. (2020) treinou um modelo formado por 3 redes ResNet50 em paralelo, cada uma pré-treinada em diferentes conjuntos de dados para a identificação de diferentes características da tosse. A saída dessas redes foi concatenada e o método fornece como saída a classificação binária da condição de um indivíduo. Esse modelo obteve uma AUC de 0,97 quando validado com amostras de indivíduos diagnosticados com um teste oficial, e uma sensibilidade de 100% com uma especificidade de 83,2% para assintomáticos. Por outro lado, Chaudhari (Chaudhari et al., 2020) desenvolveu um modelo de DNN, com 3 redes em paralelo, sendo uma Rede Neural Convolucional (*Convolutional Neural Network* -

[3] `https://coswara.iisc.ac.in/`

CNN) e duas Rede Neurais Artificiais (*Neural Network* - NN). A saída dessas redes são concatenadas e usadas para alimentar uma nova NN que será responsável pela classificação de pessoas infectadas pela Covid-19. Essas redes recebem como entrada o Espectrograma Log Mel e o MFCC extraído de gravações de tosse, assim como os sintomas reportados pelos indivíduos. O modelo obteve um AUC médio de 0,771, com o intervalo de confiança de 95% de (0,752 - 0,783). Por outro lado, Brown et al. (2020), treinaram classificadores como Regressão Logística, *Gradient Boosting Trees* e Máquinas de Vetores de Suporte. Estes modelos foram treinados três tarefas distintas: (i) classificar corretamente pacientes saudáveis e infectados com a Covid-19, (ii) distinguir um indivíduo que testou positivo para a Covid-19 e possui tosse como sintoma de um indivíduo saudável com tosse, e (iii) diferenciar a tosse de pessoas com Covid-19 daquelas com asma. Os pesquisadores reportam que o modelo conseguiu obter uma AUC maior que 0,8 em todas as tarefas.

Neste trabalho utilizou-se como base o estudo desenvolvido por Chaudhari et al. (2020), pois o modelo foi disponibilizado no Github[4] pelos autores, assim como as bases de dados utilizadas para treinamento[5,6]. Foram propostas modificações no modelo base, o que resultou em um modelo mais simples(i.e. com menor número de parâmetros a ser ajustado), e capaz de obter melhores resultados, conforme mostrado pelos experimentos computacionais realizados. Além disso, diferente dos outros trabalhos da literatura, foi proposto um método de aumento de dados para o tratamento do desbalanceamento dos dados discutido no Apêndice A.

[4] `https://github.com/virufy/virufy-covid`
[5] `https://zenodo.org/record/4048312`
[6] `https://github.com/iiscleap/Coswara-Data`

3 MATERIAIS E MÉTODOS

Neste estudo, foram utilizados sons de tosse e informações médicas dos pacientes coletados pelos grupos COUGHVID (Orlandic et al., 2021), que obteve amostras de tosse de indivíduos de diversos países, além da idade, sexo, localização geográfica e status da Covid-19. A coleta foi feita por meio do site[1] e disponibilizado em setembro de 2020 na plataforma zenodo[2]. Além disso, foram usados também os dados coletadas pelo Projeto Coswara (Sharma et al., 2020) do Indian Institute of Science coletados no site[3] e disponibilizados no GitHub[4]. Este projeto requer que os participantes forneçam uma gravação de sons respiratórios, sons de tosse, fonação sustentada de vogais, um exercício de contagem e condições de saúde, bem como informações médicas.

3.1 BASE DE DADOS AGREGADA

Similarmente ao que foi feito por Chaudhari (Chaudhari et al., 2020), neste trabalho os conjuntos de dados do grupo COUGHVID e do projeto Coswara foram combinados. Nesse caso, foram selecionados todos os 1575 registros da base de dados Coswara, sendo 109 contaminados e 1466 saudáveis. Apenas as gravações de tosse curta foram utilizadas. Em relação aos dados do COUGHVID, foram selecionadas apenas as amostras com $cough\_detected \geq 0,9$. Isso resultou em 441 amostras de indivíduos infectados e 5.651 saudáveis, onde foram selecionadas aleatoriamente 1000 das negativas e todas as positivas. Esse processo de *undersampling* foi feito conforme realizado na literatura. Dessa forma, totalizando 3016 amostras disponíveis para o treinamento do modelo, com 550 dados da classe positiva (18,24% da base) e 2466 da classe negativa (81,76% da base).

Por não existir um padrão nas informações médicas coletadas, já que cada projeto coleta conjuntos diferentes de dados, foi necessário definir as informações coletadas por ambas as bases para que elas fossem agregadas. A informação comum que foi utilizada para treinar nosso modelo é o registro da tosse disponível em ambos os casos e duas informações clínicas representadas por valores lógicos: (i) indica se o sujeito possui alguma condição respiratória pré-existente e (ii) revela se o paciente tem os sintomas, febre ou dores musculares.

O processo de preparação dos dados se iniciou com a agregação dos metadados (caminho do áudio da tosse, sintomas e status de contaminação do indivíduo) das amostras selecionadas em uma base única. Em seguida, para cada uma das gravações disponíveis calculou-se o Espectrograma Log Mel, e os λ primeiros *Mel Frequency Cepstral Coefficients* utilizando a biblioteca librosa (McFee et al., 2015), que se trata de um pacote para análise

[1] `https://coughvid.epfl.ch/`
[2] `https://zenodo.org/record/4048312`
[3] `https://coswara.iisc.ac.in/`
[4] `https://github.com/iiscleap/Coswara-Data`

de áudio e música disponível para a linguagem Python. Essas duas características extraídas das gravações da tosse (Espectrograma Log Mel e MFCC), bem como as informações médicas citadas anteriormente, foram utilizadas como entrada do modelo de classificação para predição de indivíduos infectados com a Covid-19.

3.2 MODELO COMPUTACIONAL

Neste trabalho foi utilizado como base o modelo de DNN proposto por Chaudhari et al. (2020) e cujo código está disponível no GitHub[5]. Este modelo consiste na combinação de três redes, duas NN e uma CNN, cada uma treinada com dados distintos de um mesmo indivíduo. As saídas dessas três redes foram concatenadas e serviram de entrada para uma nova NN cujo resultado é a probabilidade de um indivíduo estar infectado pela Covid-19. Uma das duas NNs iniciais foi treinada com os MFCCs extraídos do áudio da tosse, enquanto a outra recebeu como entrada as informações médicas do paciente. Já a CNN foi treinada com as imagens do Espectrograma Log Mel geradas a partir do áudio. A partir desse modelo inicial foram propostas algumas alterações na estrutura da rede, bem como no treinamento da mesma, com o objetivo de diminuir o *overfitting*, aumentar sua capacidade de generalização e também o desempenho na classificação de pacientes infectados.

3.2.1 REDES NEURAIS PROFUNDAS (*DEEP NEURAL NETWORK* - DNN)

Redes Neurais Artificiais (*Neural Network* - NN) foram inspiradas no cérebro humano, imitando como neurônios biológicos funcionam (IBM, 2020). NNs são formadas por nós contendo uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. A Figura 5 possui um exemplo de uma NN e uma DNN, onde os nós mais à esquerda (em vermelho) representam os nós de entrada, os amarelos representam os nós das camadas ocultas, e os mais a direita (em azul) a camada de saída. Uma DNN é uma categoria de NN formada por muitas camadas ocultas.

Em uma NN *Feedforward* as informações são transmitidas de uma camada a outra, com a saída de uma camada anterior fornecendo entrada para a próxima. Cada nó é composto pelas entradas, o peso que cada uma delas recebe, determinando assim sua importância na saída do nó, um viés e a saída definida por uma função matemática chamada função de ativação aplicada sobre os pesos e as entradas do neurônio. A Figura 6 ilustra um neurônio, onde $x_1 \dots x_n$ representam as entradas, $w_1 \dots w_n$ representam os pesos correspondentes a cada entrada, $\sum$ representa o combinador linear responsável por ponderar os valores de entrada, θ é o viés, u é o potencial de ativação que pode ser da forma $u = \sum w_i x_i + \theta$, g é a função de ativação aplicada sobre o potencial u e y é o sinal de saída que poderá ser usado como entrada para os nós de uma próxima camada. No

[5] `https://github.com/virufy/virufy-covid`

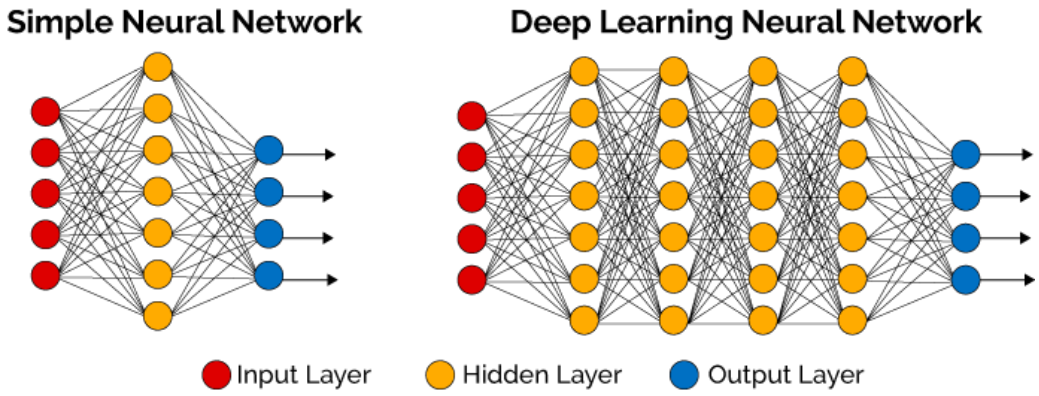


Figura 5 – Exemplo de uma Rede Neural Artificial e uma Rede Neural Profunda. Imagem retirada de Data Science Academy (2021).

modelo base utilizado nesse trabalho foram utilizadas as funções de ativação *Rectified Linear Unit* (ReLU) e *Sigmoid*. Por outro lado, no modelo modificado a função *Sigmoid* foi substituída por uma função *Softmax*.

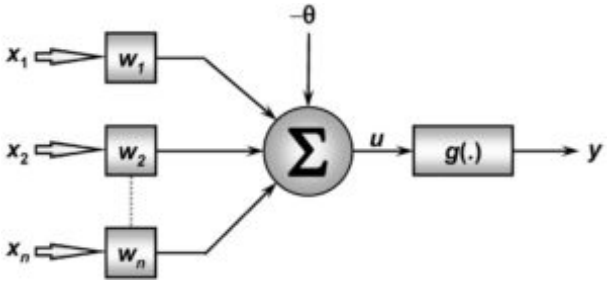


Figura 6 – Exemplo de um neurônio artificial. Imagem retirada de Data Science Academy (2021).

O processo de aprendizagem de uma NN normalmente consiste em múltiplas iterações, também chamadas épocas, onde em cada uma dessas iterações os dados de treinamento serão fornecidos ao modelo e a saída resultante será calculada. Ao final de cada época uma função de custo (ou função de perda) é calculada para avaliar o desempenho do modelo, de acordo com alguma métrica definida. Esta função servirá para ajustar os pesos e os vieses do modelo a cada iteração, com o objetivo final de minimizar a função de custo, de forma que o modelo consiga identificar os padrões presentes nos dados e generalizar o resultado para novas amostras sobre as quais ele não foi treinado. Na literatura é possível encontrar diversos algoritmos utilizados no treinamento do modelo, que serão responsáveis por atualizar os pesos do modelo. Um dos métodos mais utilizados no treinamento de DNN é o Adam (Kingma and Ba, 2014), por se tratar de um algoritmo capaz de convergir rapidamente, além de obter bons resultados.

Um problema comum que pode ocorrer em DNNs é o *overfitting*. Conforme apontado por Dietterich (Dietterich, 1995), um algoritmo de ML é treinando em um conjunto de dados de treinamento, com o objetivo de identificar os padrões presentes e ser posteriormente aplicado para fazer predições sobre novos dados. O objetivo final

é maximizar o desempenho de um determinado modelo sobre os novos dados, e não necessariamente sobre os dados de treinamento. O *overfitting* ocorre quando o modelo se adapta muito bem aos dados de treinamento, de forma que ele memorize suas peculiaridades, e perca sua capacidade de generalização para novos dados desconhecidos. Uma das técnicas que podem ser adotadas em uma DNN para diminuir o *overfitting* é chamada *dropout*, e consiste em aleatoriamente remover alguns nós das camadas ocultas durante uma época do treinamento. Dessa forma, os pesos e os vieses desses nós não serão atualizados durante aquela época. Após o fim da época esses nós são incluídos novamente na topologia da rede, e um novo conjunto de nós é sorteado para serem removidos. Esse processo é repetido durante todo o treinamento. A técnica de *dropout* foi adotada no modelo base, assim como no modelo modificado.

3.2.2 REDES NEURAIS CONVOLUCIONAIS (*CONVOLUTIONAL NEURAL NETWORK* - CNN)

Redes Neurais Convolucionais (*Convolutional Neural Network* - CNN) são uma categoria de DNN muito utilizadas na área de visão computacional. Apesar de existirem diversas aplicações possíveis, uma CNN é utilizada principalmente em tarefas de classificação de imagens e reconhecimento de objetos. Dessa forma, são normalmente utilizadas imagens como entrada do modelo. Apesar de ser possível utilizar uma DNN para classificação de imagens em alguns casos, conforme mostrado por Nielsen (Nielsen, 2015) na classificação de dígitos escritos a mão, essas redes perdem muitas informações espaciais dos píxeis de uma imagem, e não conseguem obter bons resultados em imagens com maior complexidade. O conceito de CNN conforme conhecido atualmente foi proposto por LeCun (LeCun et al., 1998). Neste trabalho, uma CNN será utilizada para classificar indivíduos infectados com a Covid-19 através de sons de tosse. Portanto, os sinais de áudio foram convertidos em Espectrogramas Log Mel, que se tratam de imagens que serão utilizadas para alimentar o modelo.

Uma CNN consegue processar imagens como tensores sendo formadas por camadas de convolução e *pooling*, seguidas por camadas totalmente conectadas conforme mostrado na Figura 7. A camada de convolução é composta por diferentes *kernels* (também chamados filtros) que possuem a capacidade extrair informações das imagens. Nas camadas iniciais o filtro será responsável por extrair características mais básicas tais como as arestas de uma imagem, mas em níveis mais profundos é possível obter informações mais complexas. Já a camada de *pooling* é responsável por reduzir o tamanho das características obtidas na camada de convolução, com o objetivo de reduzir a complexidade das camadas posteriores e diminuir o custo computacional necessário para processar os dados. Uma CNN pode conter várias camadas de convolução e de *pooling* seguidas. Ao fim desse processo de extração de informações a saída da última camada é então achatada, transformando-se em um vetor de informações que serão alimentadas em uma DNN que será responsável

pela classificação da imagem em uma das classes existentes. Esse processo é explicado com mais detalhes por Albawi Albawi et al. (2017).

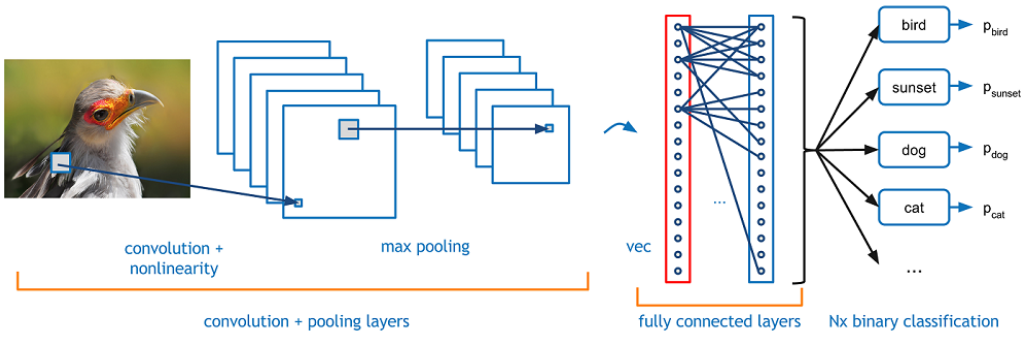


Figura 7 – Exemplo de uma Rede Neural Convolucional. Imagem retirada de Data Science Academy (2021).

3.2.3 MODELO BASE

A Figura 8 contém uma representação do modelo usado como base neste trabalho, proposto por Chaudhari et al. (2020). As duas NNs são compostas por duas camadas ocultas com uma função de ativação ReLU, seguida por uma camada de *dropout*. A primeira NN recebe como entrada as informações médicas dos pacientes, que será transmitida através da camada Totalmente Conectada (*Fully Connected* - FC), FC1 que contém 16 neurônios e uma taxa de *dropout* de 0,4 e FC2 com 64 nós com uma taxa de *dropout* de 0,2. A segunda rede é uma CNN, alimentada com as imagens do Espectrograma Log Mel da tosse com as dimensões (64, 64, 1) como entrada. Esta é formada por três camadas de convolução 2D com *kernels* de tamanho de 3 x 3, onde a camada CL1 é formada por 32 *kernels* com um *stride* de tamanho 2, enquanto ambas as camadas CL2 e CL3 consistem de 64 *kernels* com um *stride* de 1. Cada uma dessas camadas de convolução é seguida por uma camada de *average pooling 2D* com um *kernel* de tamanho 2 x 2 e um *stride* de 2, uma camada de *batch normalization* e ativação ReLU. A saída dessas camadas de convolução é achatada e usada para alimentar a camada densa FC5 composta por 256 nós com ativação ReLU e tem uma taxa de *dropout* de 0,5. A última DNN recebe como entrada os primeiro λ MFCCs extraídos do som de tosse sendo formada por duas camadas densas semelhantes à primeira, mas a camada FC3 é formada por 256 neurônios e uma taxa de *dropout* de 0,4 e a camada FC4 por 128 nós e 0,2 de *dropout*. Finalmente, as saídas dessas redes são concatenadas, e alimentadas através das duas camadas densas FC6 com 64 nós e FC7 com 32, ambas com ativação ReLU, e combinadas em um nó final com ativação *sigmoid* que prevê a probabilidade de um sujeito estar infectado com a Covid-19.

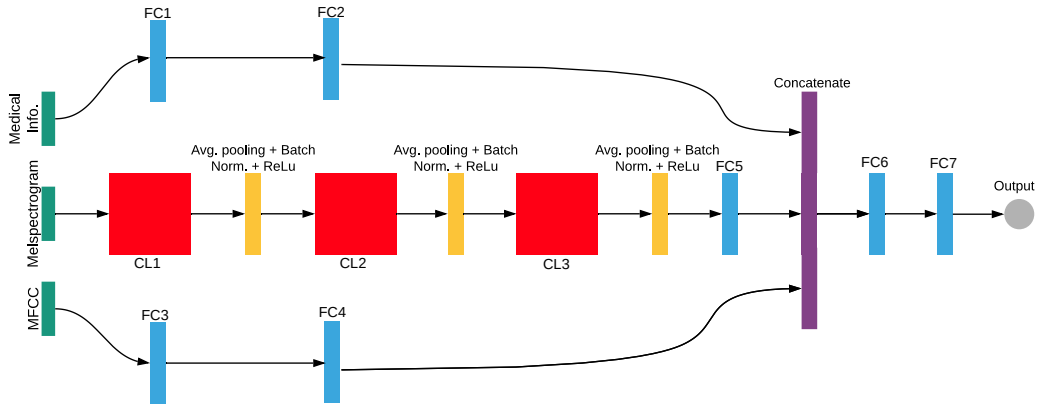


Figura 8 – Representação da Rede Neural Artificial utilizada para classificação de pacientes infectados pela Covid-19.

3.2.4 MODELO PROPOSTO

Para melhorar o desempenho do modelo e aumentar sua capacidade de generalização, neste trabalho foram propostas modificações na DNN proposta por Chaudhari et al. (2020) e seus associados. Com o objetivo de reduzir a complexidade da rede e prevenir o *overfitting*, foram feitas alterações no número de neurônios presentes em algumas das camadas totalmente conectadas presentes no modelo. O número de neurônios nas camadas FC1 e FC3 foi alterado para 64, enquanto as camadas FC2 e FC4 passaram a ser formadas por 32 nós. Já o número de neurônios na camada FC5 foi diminuído para a metade do valor original, passando a ser 128. Além disso, a saída do modelo foi alterada, de um nó com ativação *sigmoid*, para dois nós com ativação *softmax*. Com essas alterações o número de parâmetros no modelo reduziu de 279.826 no modelo original para 154.370 na versão proposta, o que representa uma redução de cerca de 45% na quantidade de parâmetros a serem ajustados.

4 EXPERIMENTOS COMPUTACIONAIS

Neste capítulo são apresentados os experimentos computacionais realizados, as métricas utilizadas para a avaliação do modelo, e as discussões acerca dos resultados obtidos. Além disso, são apresentados os resultados obtidos pelo modelo utilizado de base da literatura (Chaudhari et al., 2020), chamado aqui simplesmente de Modelo Base (MB). Para cada abordagem proposta os experimentos computacionais foram repetidos 30 vezes, com diferentes divisões aleatórias dos dados em conjuntos de treinamento, validação e teste. Foi utilizado uma divisão 70, 15 e 15, assim como na literatura. Esta divisão foi feita respeitando as proporções entre as classes no conjunto de dados original. O número de amostras para cada classe nesses conjuntos é apresentado na Tabela 1.

Tabela 1 – Distribuição dos dados nos conjuntos de treinamento, validação e teste.

	Treinamento	**Validação**	**Teste**
Covid-19 positivo	385	83	82
Covid-19 negativo	1726	370	370

Os experimentos foram realizados utilizando o conjunto de dados original gerado a partir da combinação das bases COUGHVID e Coswara. Nesse caso, foram extraídos os $\lambda = 39$ primeiros MFCC. As discussões acerca dos resultados do conjunto com aumento de dados se encontram no Apêndice A. Foram realizados experimentos com o objetivo de entender o impacto das informações passadas pelo modelo modificado (Espectrograma Log Mel, MFCC e sintomas). Inicialmente, foram realizados experimentos com o modelo completo (chamado M1). Em seguida, foram realizados novos experimentos onde uma das entradas era desconsiderada, por exemplo, o modelo M2 é composto pelas redes que recebem o Espectrograma Log Mel e os sintomas como entrada, enquanto o M3 indica que a NN é alimentada com os sintomas e o MFCC, e o modelo M4 é composto pelas redes com o MFCC e o Espectrograma Log Mel. A Tabela 2 resume as informações de composição dos modelos.

Tabela 2 – Resumo da estrutura dos modelos.

Nome do Modelo	**Descrição**
M1	Modelo completo.
M2	Modelo formado pelas redes alimentadas com o Espectrograma Log Mel e os sintomas.
M3	Modelo formado pelas redes alimentadas com o MFCC e os sintomas.
M4	Modelo formado pelas redes alimentadas com o Espectrograma Log Mel e o MFCC.

Foi utilizado no treinamento do modelo a entropia cruzada categórica como função de perda, e um otimizador Adam com uma taxa de aprendizagem de 0,0001. Os dados foram divididos em *batches* de tamanho 32 e o modelo foi treinado por 50 épocas. Como forma de regularização para evitar o *overfitting* e a degradação da generalização do modelo, durante o treinamento, foi implementada uma política de *Early Stopping*(ES). O ES Prechelt (1998) é uma técnica utilizada para monitorar a desempenho do modelo através do conjunto de validação, com o intuito de evitar que o desempenho da rede piore no conjunto monitorado, enquanto melhora no de treinamento. Neste estudo foi implementado o ES monitorando a AUC dos dados de validação, com uma paciência limite de 20 épocas.

O modelo, assim como o pré-processamento dos dados, foram implementados utilizando a linguagem de programação *Python*. Os experimentos foram executados no ambiente de desenvolvimento Google Colab[1], e o modelo foi treinado com uma placa de vídeo Nvidia Tesla P100. As NNs foram implementadas e treinadas utilizando a biblioteca *Keras*.

4.1 MÉTRICAS UTILIZADAS

Como forma de avaliar o desempenho dos modelos, na seção de resultados serão apresentadas as médias e os intervalos de confiança de 95% das seguintes métricas:

- Área sob a curva ROC (*Area Under the ROC Curve* - AUC) (Sarang Narkhede, 2018): Indica a capacidade do modelo em distinguir entre classes, sendo baseado na curva ROC, que se trata de uma curva probabilística que traça a taxa de verdadeiros positivos pela taxa de falsos positivos em diferentes limiares. O valor do AUC varia entre 0,0, cujas previsões do modelo estão todas erradas, até 1,0 cujas previsões estão todas corretas.

- Sensibilidade (Trevethan, 2017): A sensibilidade indica a capacidade do modelo em classificar corretamente as amostras da classe positiva, e pode ser calculada como $Sens = \frac{VP}{VP+FN}$. Onde VP representa as amostras da classe positiva classificadas corretamente e FN as amostras classificadas como negativas.

- Especificidade (Trevethan, 2017): A sensibilidade indica a capacidade do modelo em classificar corretamente as amostras da classe negativa, e pode ser calculada como $Esp = \frac{VN}{VN+FP}$. Onde VN representa as amostras da classe negativas classificadas corretamente e FN as amostras classificadas como positivas.

- Valor preditivo positivo (VPP) (Trevethan, 2017): O VPP indica a taxa de predições positivas que realmente fazem parte da classe positiva, e pode ser calculada como

[1] `https://colab.research.google.com/`

$VPP = \frac{VP}{VP+FP}$. Onde VP representa as amostras da classe positivas classificadas corretamente e FP as amostras negativas classificadas como positivas.

- Valor preditivo negativo (VPN) (Trevethan, 2017): O VPN indica a taxa de predições negativas que realmente fazem parte da classe negativa, e pode ser calculada como $VPN = \frac{VN}{VN+FN}$. Onde VN representa as amostras da classe negativa classificadas corretamente e FN as amostras positivas classificadas como negativas.

No estudo executado por Chaudhari et al. (2020), a única métrica apresentada para o modelo é o AUC médio e o intervalo de confiança de 95% para 5 execuções do modelo, considerando diferentes divisões aleatórias dos dados nos conjuntos de treinamento, validação e teste. Neste estudo, além do AUC serão apresentados os resultados obtidos pelo modelo segundo as métricas apresentadas anteriormente. O AUC é uma métrica importante, pois indica a capacidade do modelo de diferenciar entre as classes. Logo, quanto maior o AUC melhor a capacidade do método de distinguir entre indivíduos infectados e saudáveis. Além disso, o VPP e o VPN são métricas importantes para medir o desempenho do modelo na predição correta das amostras na classe positiva e negativa, respectivamente.

4.2 RESULTADOS OBTIDOS

Os resultados obtidos no treinamento do modelo proposto, assim como as variações do mesmo, onde uma das entradas da rede foi desconsiderada são apresentados na Tabela 3. Esta tabela contém os resultados obtidos no conjunto de teste, para os métodos.

Tabela 3 – A média ($\bar{X}$) e o intervalo de confiança (IC) de 95% obtido pelos modelos propostos para as métricas utilizadas.

	AUC		**Sensibilidade**		**Especificidade**		**VVP**		**VPN**	
	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**
M1	0,88	(0,88, 0,89)	**0,82**	(0,81, 0,82)	0,95	(0,94, 0,96)	**0,82**	(0,81, 0,82)	0,84	(0,84, 0,85)
M2	**0,89**	(0,88, 0,89)	**0,82**	(0,82, 0,83)	0,96	(0,96, 0,97)	**0,82**	(0,82, 0,83)	0,84	(0,84, 0,85)
M3	0,87	(0,87, 0,88)	**0,82**	(0,82, 0,83)	0,96	(0,95, 0,97)	**0,82**	(0,82, 0,83)	**0,85**	(0,84, 0,85)
M4	0,85	(0,84, 0,85)	0,80	(0,80, 0,81)	**0,97**	(0,96, 0,98)	0,80	(0,80, 0,81)	0,82	(0,82, 0,82)
MB	0,77	(0,74, 0,80)	—	—	—	—	—	—	—	—

Considerando o AUC médio e o intervalo de confiança de 95%, é possível concluir que as alterações propostas conseguiram gerar modelos com maior capacidade de classificação de indivíduos infectados pela Covid-19, já que todos os modelos propostos, independente do conjunto de treinamento, obtiveram um AUC médio maior que o modelo MB. Assim como o intervalo de confiança que no pior caso ainda é superior ao resultado da literatura e sem a sobreposição de valores. Comparando o AUC médio do modelo M2 com o resultado da literatura, é possível perceber que este modelo obteve um resultado cerca de 16% melhor.

Os resultados do AUC indicam que o modelo M2, alimentado com o Espectrograma Log Mel e os sintomas do paciente, possui a maior capacidade de distinção entre um indivíduo infectado e um saudável. Portanto, do ponto de vista da classificação este é o modelo com melhor desempenho dentre os propostos. Entretanto, do ponto de vista médico o modelo M3 é o que apresenta as maiores taxas de classificação correta de indivíduos na classe positiva (infectados) e na classe negativa (saudáveis), conforme indicado pelas métricas VPP e VPN. Portanto, este é o modelo com as maiores taxas de acerto nas classes existentes, o que representa uma menor taxa de falsos negativos e falsos positivos. A taxa de falsos negativos é muito importante, pois uma amostra de um indivíduo infectado marcado como negativo pelo modelo, pode representar uma pessoa potencialmente espalhando o vírus para outros. Além disso, é possível observar que o modelo M4 possui o pior resultado na classificação de pessoas infectadas com Covid-19. Dessa forma, é possível perceber que os sintomas de um paciente possuem uma grande contribuição na capacidade preditiva do modelo, e quando desconsiderado, diminui o desempenho do modelo.

A Figura 9 apresenta a curva ROC e o AUC médio obtido e pelos modelos.

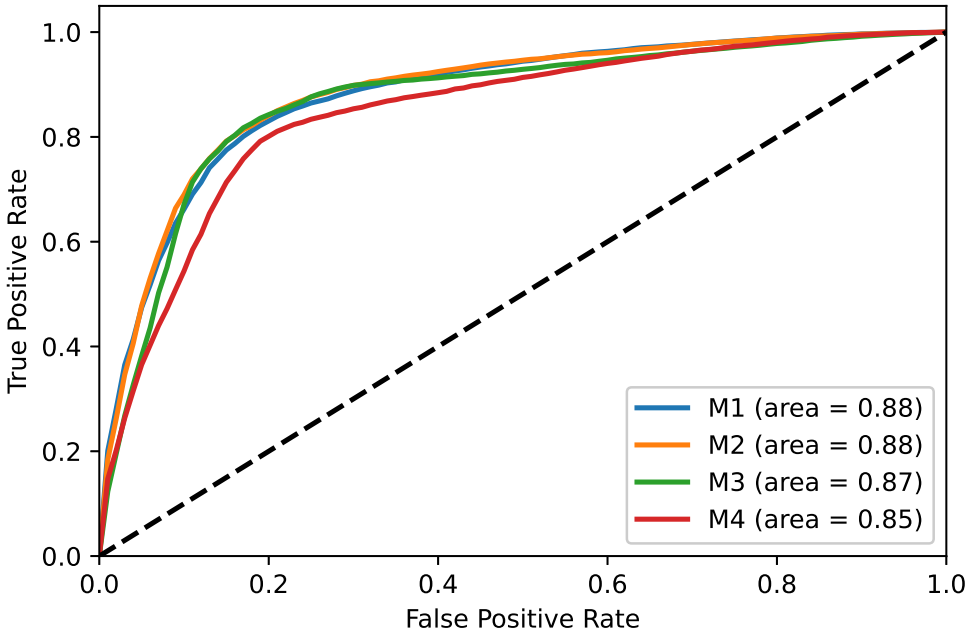


Figura 9 – Curva ROC e AUC médio dos modelos.

5 CONCLUSÕES E TRABALHOS FUTUROS

Detecção de doenças através de sons de tosses forçadas é um campo de pesquisa recente, e ganhou muito destaque com surgimento da pandemia do novo coronavírus em 2020. Este estudo foi baseado em um modelo de Redes Neurais Profundas disponível na literatura e foram propostas aqui algumas alterações em sua estrutura, assim como no protocolo de treinamento do mesmo. Para o treinamento deste método, foram utilizadas dados das bases COUGHVID e Coswara, disponíveis na literatura. Essas bases são compostas por gravações de tosse forçada e informações clínicas, sendo combinadas em um conjunto único de dados. Para tratar o grande desbalanceamento dos dados, com 81,76% pertencendo à classe negativa, foi proposta uma técnica de aumento de dados aplicada no conjunto de treinamento. Além disso, com o objetivo de entender a contribuição de cada uma das entradas fornecidas ao algoritmo proposto foi avaliado o desempenho do modelo quando uma das entradas era desconsiderada. O objetivo de propor um método com melhor desempenho na detecção de pacientes infectados com Covid-19, foi atingido neste trabalho. Entretanto, para que o mesmo possa ser utilizado pela população geral é necessário que mais estudos sejam realizados.

Para avaliação do modelo foram apresentadas as médias e o intervalo de confiança de 95% da AUC, Sensibilidade, Especificidade, Valor Preditivo Positivo (VPP) e Valor Preditivo Negativo (VPN). Os resultados comprovam que o modelo proposto é superior ao da literatura, já que todos os experimentos obtiveram um AUC médio maior que o modelo base, assim como, os intervalos de confiança melhores. O aumento de dados como proposto diminuiu a capacidade de generalização dos métodos. Além disso, foi possível perceber que o modelo M2 considerando apenas o Espectrograma Log Mel e os sintomas de um indivíduo obteve os melhores resultados na classificação de indivíduos infectados. Entretanto, o modelo que obteve as maiores taxas de acerto para cada uma das classes foi o modelo M3, alimentado com o MFCC e os sintomas de um paciente. Neste trabalho, foram propostas modificações em um modelo presente na literatura, e os resultados comprovam que o modelo proposto possui um desempenho superior ao método utilizado como base. Além disso, foi realizado um estudo de uma técnica de aumento de dados para gravações de tosse forçada, e analisado o impacto de cada uma das entradas, na classificação de pessoas infectadas pela Covid-19.

Apesar dos resultados obtidos neste trabalho, para que este modelo possa ser utilizado para detecção de pessoas infectados com Covid-19, é necessário que mais estudos sejam desenvolvidos. Ainda existem limitações no entendimento se o modelo consegue identificar características na tosse específicas da Covid-19, ou se ele apenas detecta uma anomalia na mesma. Nesse último caso, outras doenças que possuem tosse como sintoma também seriam classificadas como Covid-19. Além disso, é necessário entender como a idade, o gênero, o local de nascimento, e outras características podem influenciar no

desempenho do modelo.

Como trabalhos futuros, podem ser realizados novos estudos sobre o aumento de dados aplicado à sons de tosse forçada, dado que muitas das bases disponíveis possuem dados desbalanceados. Estes estudos podem envolver a utilização de outros operadores, além da busca pela melhor combinação de parâmetros dos mesmos. Outra possibilidade está no aumento de dados do espectrograma, conforme vêm sendo feito em aplicações de Reconhecimento Automático de Fala. Com o objetivo de entender a capacidade do modelo de identificação de características presentes na tosse de um indivíduo com Covid-19, um estudo pode ser realizado utilizando uma base de dados construída com áudios de indivíduos infectados com a Covid-19, assim como indivíduos que possuam outras doenças cuja tosse é apresentada como sintoma. Dessa forma, será possível observar se o modelo consegue extrair características específicas para a identificação da Covid-19. Ainda, é possível testar o desempenho do modelo em outra base de dados, não utilizadas no treinamento do mesmo, para verificar a capacidade de generalização do modelo para dados nunca antes vistos.

A AUMENTO DE DADOS

A.1 TRATAMENTO DO DESBALANCEAMENTO DOS DADOS

A combinação dos conjuntos de dados resultou em um número total de 3.016 amostras, sendo 2.466 de indivíduos com Covid-19 negativo e apenas 550 daqueles infectados, representando um problema de classificação altamente desequilibrado, conforme mostrado na Tabela 4. Um tópico muito comum e amplamente discutido em problemas de classificação é o desbalanceamento dos dados, onde normalmente a maioria das instâncias pertence a uma determinada classe, enquanto uma minoria a outra classe, normalmente a mais importante. Nesses casos é usual perceber que o modelo tende a classificar a maioria das amostras como sendo da classe predominante. Por se tratar de um problema tão comum, é possível encontrar na literatura diversos estudos (Kotsiantis et al., 2006; Guo et al., 2008; Schlüter and Grill, 2015; Nanni et al., 2020) com diferentes técnicas para atenuar este problema. Algumas das técnicas mais comuns são: (i) *undersampling*, que consiste na eliminação randômica de amostras da classe majoritária; (ii) *oversampling*, onde amostras da classe minoritária são randomicamente selecionadas e replicadas; (iii) *data augmentation*, consiste na criação de dados sintéticos ou ligeiramente modificadas a partir de existentes, conforme discutido por Dimitri Kotsiantis et al. (2006) e Xinjian (Guo et al., 2008).

Tabela 4 – Distribuição dos dados nas bases de dados Coughvid e Coswara.

	Status de Covid-19 do indivíduo		
Base de Dados	**Positivo (%)**	**Negativo (%)**	**# Amostras**
COUGHVID	441	1000	1441
Coswara	109	1466	1575
Total	550 (18,24%)	2466 (81,76%)	3016

Para diminuir o desbalanceamento dos dados e mitigar os problemas causados no treinamento do modelo, foram aplicadas técnicas de *data augmentation* nos dados relacionados a gravações de tosse e *oversampling* nas informações de sintomas das pessoas. Portanto, neste trabalho foi desenvolvido um modelo híbrido com um processo de *undersampling* na junção das bases de dados, e um *oversampling* e *data augmentation* para balanceamento dos dados. Na literatura é possível encontrar estudos de aumento dos dados em aplicações que utilizam dados de áudio como entrada (Nanni et al., 2020; Schlüter and Grill, 2015), tais como: reconhecimento de fala, classificação de animais e ambientes pelo som, músicas, entre outros. Entretanto, o aumento de dados em aplicações que utilizam sons de tosse é uma área de pesquisa pouco explorada, portanto, não se sabe ao certo qual impacto estas técnicas podem causar no resultado do modelo.

Antes da fase de aumento, os dados foram divididos em conjuntos de treinamento, validação e teste respeitando a proporção entre as classes no conjunto de dados original. O número de amostras para cada classe nesses conjuntos é apresentado na Tabela 1. Em seguida, foram aplicadas as técnicas de *data augmentation* e *oversampling* na classe Covid-19 positivo dos dados de treinamento, até que o número de amostras de pacientes infectados e saudáveis fossem iguais. Neste trabalho foram utilizados 4 operadores distintos responsáveis por diferentes transformações no áudio, e o processo de *data augmentation* foi aplicado sobre as gravações de tosse originais fornecidas pelos voluntários. Os operadores utilizados são:

- ***Gaussian Noise***: Adiciona um ruído estático em toda a amostra com uma distribuição normal no domínio temporal. A amplitude do ruído adicionado ao sinal de entrada é selecionado aleatoriamente na faixa de [0,001; 0,01].
- ***Time Strech***: Consiste na alteração da velocidade do sinal sem alteração do tom. Esta alteração na velocidade do sinal é selecionada aleatoriamente na faixa de [0,8; 1,2].
- ***Pitch Shift***: Trata-se de uma alteração no tom do sinal, podendo variar na faixa de [-2; 2] semitom.
- ***Shift***: Corresponde ao deslocamento do áudio no tempo, alterando a posição do sinal. É um valor selecionado aleatoriamente na faixa de [-0,2; 0,2].

Considerando que não existem muitos trabalhos que estudam o efeito da aplicação desses operadores em gravações de tosse na detecção de doenças, a faixa ideal de intensidade de aplicação para cada um desses operadores é desconhecida. Dessa forma, a faixa definida para os operadores *Time Stretch* e *Pitch Shift* foi baseada em estudo desenvolvido por Nanni (Nanni et al., 2020) na aplicação de *data augmentation* em áudios de animais. Para os demais operadores *Gaussian Noise* e *Shift* a faixa de variação foi definida empiricamente, baseado em testes feitos com variações dos valores apresentados no caso de uso disponível na documentação da biblioteca audiomentations[1]. O estudo dessas faixas de valores, bem como a aplicação de outros operadores é um campo que deve ser explorado em trabalhos futuros.

A metodologia para aumento dos dados consistiu na seleção aleatória de uma amostra de áudio da classe Covid-19 positiva. Em seguida, um dos operadores é selecionado aleatoriamente e aplicado sobre a amostra com uma probabilidade γ e uma intensidade θ. Essa intensidade θ é definida aleatoriamente na faixa de valores mostrada anteriormente para cada operador. Caso o operador não seja aplicado o processo é interrompido. Esse processo, descrito no Algoritmo 1, se repete até que um operador não seja aplicado ou que

[1] `https://pypi.org/project/audiomentations/`

todos tenham sido utilizados. Dessa forma, mesmo que um áudio seja sorteado mais de uma vez para criação de uma nova amostra sintética, o resultado irá variar dependendo dos operadores, da ordem e da intensidade em que foram aplicados. Para cada nova amostra sintética de áudio criada pelo processo descrito anteriormente, os sintomas associados ao sinal original foram replicados.

Algoritmo 1 Aumento de dados

```
operators ← [gaussian, time_stretch, shift, pitch_shift]
instance ← getRandom(dataset)
newInstance ← instance
op ← removeRandomOperador(operators) with probability γ
while operators is not empty AND op != null do
    newInstance ← op(newInstance, θ)
    op ← removeRandomOperador(operators) with probability γ
end while
if newInstance ! = instance then
    newDataset ← newInstance
end if
```

A.2 EXPERIMENTOS COMPUTACIONAIS E RESULTADOS OBTIDOS

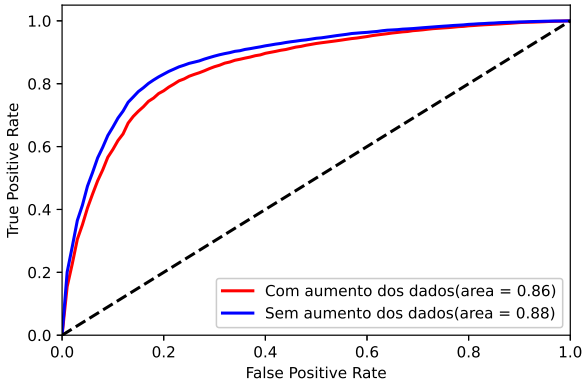
Os experimentos foram realizados conforme explicado na Seção 4. A probabilidade γ utilizada no aumento dos dados foi de 50%. O processo de aumento dos dados ocorreu no conjunto de treinamento após o processo de divisão, portanto, foi necessário gerar dados aumentados diferentes para cada um dos conjuntos de treinamento. Os resultados obtidos no treinamento do modelo proposto, assim como as variações do mesmo, onde uma das entradas da rede foi desconsiderada são apresentados na Tabela 5. Esta tabela contém os resultados obtidos no conjunto de teste, para os métodos treinados com e sem aumento dos dados.

Tabela 5 – A média ($\bar{X}$) e o intervalo de confiança (IC) de 95% obtido pelos modelos propostos para as métricas utilizadas, considerando o conjunto original e o conjunto com aumento de dados.

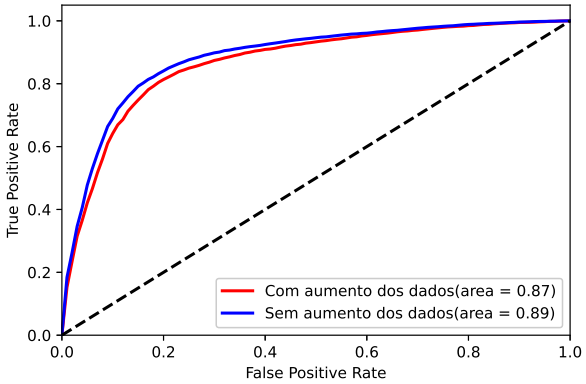
		AUC		**Sensibilidade**		**Especificidade**		**VVP**		**VPN**	
		$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**	$\bar{X}$	**95% IC**
Sem Aumento dos Dados	**M1**	0,88	(0,88, 0,89)	**0,82**	(0,81, 0,82)	0,95	(0,94, 0,96)	**0,82**	(0,81, 0,82)	0,84	(0,84, 0,85)
	M2	**0,89**	(0,88, 0,89)	**0,82**	(0,82, 0,83)	0,96	(0,96, 0,97)	**0,82**	(0,82, 0,83)	0,84	(0,84, 0,85)
	M3	0,87	(0,87, 0,88)	**0,82**	(0,82, 0,83)	0,96	(0,95, 0,97)	**0,82**	(0,82, 0,83)	**0,85**	(0,84, 0,85)
	M4	0,85	(0,84, 0,85)	0,80	(0,80, 0,81)	**0,97**	(0,96, 0,98)	0,80	(0,80, 0,81)	0,82	(0,82, 0,82)
Com Aumento dos Dados	**M1**	0,86	(0,85, 0,86)	0,79	(0,78, 0,80)	0,91	(0,89, 0,93)	0,79	(0,78, 0,80)	**0,85**	(0,84, 0,85)
	M2	0,87	(0,86, 0,87)	0,81	(0,80, 0,82)	0,93	(0,92, 0,95)	0,81	(0,80, 0,82)	**0,85**	(0,84, 0,86)
	M3	0,83	(0,82, 0,85)	0,80	(0,79, 0,81)	0,93	(0,91, 0,95)	0,80	(0,79, 0,81)	0,84	(0,83, 0,85)
	M4	0,83	(0,82, 0,84)	0,78	(0,77, 0,79)	0,93	(0,91, 0,95)	0,78	(0,77, 0,79)	0,82	(0,82, 0,83)
	MB	0,77	(0,74, 0,80)	—	—	—	—	—	—	—	—

Considerando o impacto do aumento dos dados no conjunto de treinamento, é possível perceber que este causou um impacto negativo na capacidade de classificação do modelo, já que os resultados pioraram quando comparados ao mesmo modelo treinado sem o aumento. Por ser uma área ainda pouco explorada é difícil precisar o motivo deste fato ter ocorrido, já que ainda não é muito bem compreendido como o modelo é capaz distinguir o status do paciente pelos sons de tosse forçada, assim como o impacto dos operadores utilizados nos sons da tosse. Como trabalho futuro é possível conduzir um estudo mais aprofundado para entender o impacto dos operadores, assim como um ajuste dos parâmetros presentes no mesmo. Além disso, é possível investigar a aplicação de outras técnicas de aumento de dados em áudios, tais como o aumento de dados em um espectrograma, conforme proposto por pesquisadores do Google Brain (Park et al., 2019) no caso de Reconhecimento Automático de Fala.

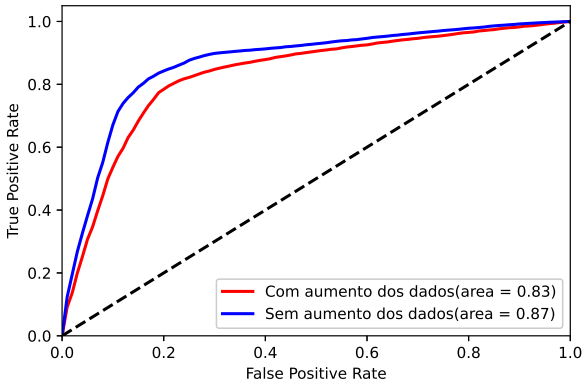
A Figura 10 apresenta também a Curva ROC e o AUC médio de um mesmo modelo, considerando seu treinamento com os dados aumentados e sem aumento. Estes gráficos reforçam o comportamento observado de piora na generalização do modelo quando treinados com os dados aumentados.



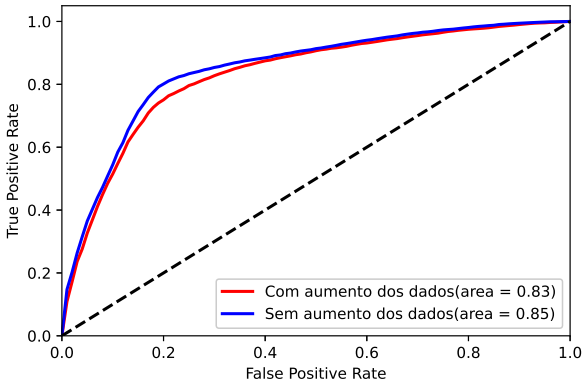
(a) Modelo M1



(b) Modelo M2



(c) Modelo M3



(d) Modelo M4

Figura 10 – Curva ROC e AUC médio dos modelos considerando o conjunto de treinamento com a presença de dados aumentados, e sem dados aumentados.

REFERÊNCIAS

Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6. Ieee, 2017.

GHR Botha, G Theron, RM Warren, M Klopper, K Dheda, PD Van Helden, and TR Niesler. Detection of tuberculosis by automatic cough sound analysis. *Physiological measurement*, 39(4):045005, 2018.

Diogo Braga, Ana M Madureira, Luis Coelho, and Reuel Ajith. Automatic detection of parkinson's disease based on acoustic analysis of speech. *Engineering Applications of Artificial Intelligence*, 77:148–158, 2019.

Chloë Brown, Jagmohan Chauhan, Andreas Grammenos, Jing Han, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, and Cecilia Mascolo. Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3474–3484, 2020.

Andrés Camero, Jamal Toutouh, and Enrique Alba. Low-cost recurrent neural network expected performance evaluation. *Preprint submitted to Elsevier March 12, 2019. arXiv:1805.07159*, 05 2018.

Gunvant Chaudhari, Xinyi Jiang, Ahmed Fakhry, Asriel Han, Jaclyn Xiao, Sabrina Shen, and Amil Khanzada. Virufy: Global applicability of crowdsourced and clinical datasets for ai detection of covid-19 from cough. *arXiv preprint arXiv:2011.13320*, 2020.

Data Science Academy. Deep Learning Book. `https://www.deeplearningbook.com.br/`, 2021. Online; Acessado em 08 de agosto de 2021.

Tom Dietterich. Overfitting and undercomputing in machine learning. *ACM computing surveys (CSUR)*, 27(3):326–327, 1995.

Alan R Fachini and Milton Roberto Heinen. Aplicação de mfcc para modelar sons de instrumentos musicais. In *11th proceedings of brazilian congress on computational intelligence*, page 31, 2016.

Xinjian Guo, Yilong Yin, Cailing Dong, Gongping Yang, and Guangtong Zhou. On the class imbalance problem. In *2008 Fourth International Conference on Natural Computation*, volume 4, pages 192–201, 2008. doi: 10.1109/ICNC.2008.871.

IBM. Neural Networks. `https://www.ibm.com/cloud/learn/neural-networks`, 2020. Online; Acessado em 08 de agosto de 2021.

Dan Jurafsky and James H. Marin. *An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall, 2008.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas, et al. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, 30(1):25–36, 2006.

Jordi Laguarta, Ferran Hueto, and Brian Subirana. Covid-19 artificial intelligence diagnosis using only cough recordings. *IEEE Open Journal of Engineering in Medicine and Biology*, 1:275–281, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25. Citeseer, 2015.

Loris Nanni, Gianluca Maguolo, and Michelangelo Paci. Data augmentation approaches for improving animal audio classification. *Ecological Informatics*, 57:101084, 2020.

Michael A Nielsen. *Neural networks and deep learning*, volume 25. Determination press San Francisco, CA, 2015.

Lara Orlandic, Tomas Teijeiro, and David Atienza. The coughvid crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms. *Scientific Data*, 8(1):1–10, 2021.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 1998.

Sarang Narkhede. Understanding AUC - ROC Curve. `https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5`, 2018. Online; Acessado em 08 de agosto de 2021.

Jan Schlüter and Thomas Grill. Exploring data augmentation for improved singing voice detection with neural networks. In *ISMIR*, pages 121–126, 2015.

Neeraj Sharma, Prashant Krishnan, Rohit Kumar, Shreyas Ramoji, Srikanth Raj Chetupalli, Prasanta Kumar Ghosh, Sriram Ganapathy, et al. Coswara–a database of breathing, cough, and voice sounds for covid-19 diagnosis. *arXiv preprint arXiv:2005.10548*, 2020.

R Sonu and K Sharma. Disease detection using analysis of voice parameters. *International Journal of Computing Science and Communication Technologies*, 4(2):416420, 2012.

Stanley S Stevens and John Volkmann. The relation of pitch to frequency: A revised scale. *The American Journal of Psychology*, 53(3):329–353, 1940.

Robert Trevethan. Sensitivity, specificity, and predictive values: foundations, pliabilities, and pitfalls in research and practice. *Frontiers in public health*, 5:307, 2017.

Maxim Vashkevich, Alexander Petrovsky, and Yuliya Rushkevich. Bulbar als detection based on analysis of voice perturbation and vibrato. In *2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pages 267–272. IEEE, 2019.