

## AUTOENCODER VARIACIONAL COMO TÉCNICA DE BALANCEAMENTO DE BANCO DE DADOS DE ÁUDIOS RESPIRATÓRIOS

### *VARIATIONAL AUTOENCODER AS A BALANCING TECHNIQUE FOR RESPIRATORY SOUND DATABASE*

Vitoriano Medeiros Casas<sup>1</sup>  
Daniel Akio Chen<sup>2</sup>  
João Paulo Campos Mendes<sup>3</sup>  
Moisés Pereira Bastos<sup>4</sup>  
Victor Vermehren Valenzuela<sup>5</sup>  
César Augusto Menezes Nepomuceno<sup>6</sup>

**Resumo** – *Quando se trabalha com inteligência artificial, idealmente, um banco de dados balanceado é necessário (quando os dados estão aproximadamente em quantidades iguais), no entanto, nem sempre esse balanceamento é possível. Por essa razão, modificar a base de dados para se trabalhar com dados balanceados é uma alternativa válida para lidar com esse problema. Esse trabalho realizou a codificação pela linguagem python, testes e validação do Variational Autoencoder (VAE), um algoritmo de Machine Learning que tem o objetivo de gerar imagens a partir de exemplos anteriores. Treinando o VAE com imagens de mel-espectrogramas gerados a partir de áudios previamente adquiridos, foi possível criar novas imagens com sucesso. Graças a isso, foi viabilizado o aumento da quantidade de elementos na base de dados e um consequente balanceamento do banco. Isso foi importante para que, em trabalhos futuros, seja possível fazer a criação de modelos mais eficientes de Machine Learning de classificação.*

**Palavras-chave:** *Aprendizado de Máquina. Rede Neural. Autoencoder Variacional.*

**Abstract** - *When working with artificial intelligence, ideally, a balanced database (when the data is in about equal amounts) is required, however, this is often not always possible. For this*

<sup>1</sup> Graduando em Engenharia de Controle e Automação (UEA). Contato: [vmc.eai18@uea.edu.br](mailto:vmc.eai18@uea.edu.br).

<sup>2</sup> Graduando em Engenharia da Computação (UEA). Contato: [dac.eng17@uea.edu.br](mailto:dac.eng17@uea.edu.br).

<sup>3</sup> Mestre em Engenharia Elétrica (UFAM). Docente no curso de Engenharia da Computação (UEA). Contato: [jpcmendes@uea.edu.br](mailto:jpcmendes@uea.edu.br).

<sup>4</sup> Mestre em Ciências na área de Sistemas Aeroespaciais e Mecatrônica (ITA). Professor Assistente no curso de Engenharia de Controle e Automação (UEA). Contato: [mpbastos@uea.edu.br](mailto:mpbastos@uea.edu.br).

<sup>5</sup> Doutor em Engenharia Elétrica (UFPE/UEA). Professor Adjunto no curso de Engenharia Elétrica (UEA). Contato: [vvalenzuela@uea.edu.br](mailto:vvalenzuela@uea.edu.br).

<sup>6</sup> Graduando em Informática (UEA). Contato: [camn.lic@uea.edu.br](mailto:camn.lic@uea.edu.br).

*reason, modifying the database to work with balanced data is a valid alternative to deal with this problem. This work performed the coding, testing and validation of the Variational Autoencoder (VAE), a Machine Learning algorithm that aims to generate images from previous examples. By training the VAE with mel-spectrogram images generated from previously acquired audios, it was possible to successfully create new images. Thanks to this, it was made possible to increase the amount of elements in the database and a consequent balancing of the database. This was important so that, in future work, it will be possible to create more efficient Machine Learning models for classification.*

**Keywords:** *Machine Learning. Neural Network. Variational Autoencoder.*

## I. INTRODUÇÃO

Datado desde 1679 e descrito pelo comitê americano da sociedade torácica em 1962 (MANDAL, 2019), o DPOC (Doença Pulmonar Obstrutiva Crônica) é uma doença respiratória caracterizada pela diminuição das vias aéreas, o que pode causar falta de ar e tosse. O DPOC é uma patologia de difícil detecção e que apresenta sintomas os quais são muito comuns a outras doenças, como a Asma. É uma doença progressiva que se agrava com o tempo e pode causar a morte se não houver tratamento (VESTBO, 2013). Devido a isso, é a terceira doença pulmonar que mais mata pessoas no mundo (WORLD HEALTH ORGANIZATION, 2017). Doenças respiratórias, como a DPOC, podem ser causadas por fatores naturais como ambiente seco, infecções virais ou até mesmo genéticas, no entanto, a maior causa dessa enfermidade é a poluição do ar e o tabagismo (THUN *et al.*, 2013). Elas geram bastante preocupação devido ao número de pessoas atingidas por essas enfermidades, que pode chegar a 65 milhões de pessoas apenas em DPOC (FÓRUM INTERNACIONAL DE SOCIEDADES RESPIRATÓRIAS, 2017). Essa doença respiratória teve um grande impacto econômico e social no mundo (MARQUES, OLIVEIRA E JÁCOME, 2014).

Para estimular o estudo de programação e a pesquisa de doenças respiratórias, a Conferência Internacional de Informática Biomédica e Saúde (ICBHI do inglês International Conference on Biomedical and Health Informatics), em parceria com a Universidade de Aveiro (ESSUA), a Universidade Aristóteles de Thessaloniki (AUTH) e a Universidade de Coimbra (UC), propôs um desafio científico para desenvolvimento de algoritmos para caracterização de áudios respiratórios utilizando aprendizado de máquina (ROCHA *et al.*, 2018) e Inteligência Artificial (IA). Nesse desafio, foi disponibilizado um banco de dados recolhido pelas universidades parceiras. A base de dados conta com várias amostras de áudios coletados em ambientes clínicos e não clínicos, com o intuito de serem utilizados no treinamento de modelos de *Machine Learning*. Além do mais, o banco de dados dispõe de informações detalhadas do indivíduo do qual os áudios foram retirados, como sua idade, sexo, peso, entre outras.

Entretanto, um problema desta base é que ela apresenta desbalanceamento de classes, que é quando existem muito mais instâncias de uma classe do que de outra. Isso interfere diretamente no desempenho de modelos de aprendizado de máquina, como por exemplo a Convolutional Neural Network (CNN). Dessa forma, este trabalho propôs o estudo de uma técnica de aumento de dados utilizando Variational Autoencoder (VAE) para fazer o balanceamento dos dados. O VAE é parte de um tipo de rede neural conhecida como autoencoders (GARCIA-ÓRDAS *et al.*, 2020), sua principal funcionalidade é ser capaz de gerar novas instâncias a partir das antigas.

Além de estudar a teoria e a prática do VAE, o presente trabalho teve como objetivo fazer o estudo de *Machine Learning* e redes neurais na linguagem de programação python a qual é amplamente usada para desenvolvimento e pesquisas

científicas (RASCHKA, 2015). Contribuindo assim para o desenvolvimento na área de IA e gerando uma ferramenta de classificação de ausculta pulmonar.

## II. REFERÊNCIA TEÓRICA

### 2.1 - *Aprendizado de máquina (Machine Learning)*

Segundo Luger (2013) e Simon (1996), aprendizado de máquina pode ser definido como:

“qualquer mudança em um sistema que melhore o seu desempenho na segunda vez que ele repetir a mesma tarefa ou outra tarefa retirada da mesma população.”

*Machine Learning* é um estudo de algoritmos que, por meio do uso de dados e a estatística, é capaz de melhorar a performance de uma tarefa. Utilizando dados de treinamento, que também são chamados de dados históricos, um modelo pode ser criado para se fazer previsões e/ou decisões que não foram programadas originalmente (RASCHKA, 2015).

Essa técnica pode ser usada em qualquer área que tange a estatística, tais como a medicina, meteorologia, financeiro etc. No entanto, ao contrário da estatística, que procura inferir informações através de amostragem de uma população, o aprendizado de máquina procura achar padrões preditivos naquelas informações (BZDOK; ALTMAN; KRZYWINSKI, 2018).

### 2.2 - *Rede Neural (Neural Network)*

Redes Neurais (RN), ou também chamado de Redes Neurais Artificiais (RNA), é um sistema computacional inspirado por redes neurais biológicas que os cérebros animais possuem (GOODFELLOW; BENGIO; COURVILLE, 2016).

Esse método é uma subdivisão de *Machine Learning*, ou seja, sua principal funcionalidade é ser treinado com amostras anteriores (que possuem suas respectivas características e resultados), para que ele possa aprender como esse objeto chegou ao seu resultado, com a finalidade de que sejam geradas previsões precisas desse elemento. Para fazer isso, essa técnica utiliza-se de funções matemáticas para formar associações entre a entrada e a saída, descrita assim pelo modelo *threshold logic* (lógica limite) de McCulloch e Pitts (1943).

Assim como escreve Norvig (2013), uma rede neural é apenas uma coleção de unidades conectadas que “disparam” quando uma combinação de suas entradas ocorre, sendo as propriedades dessa rede determinadas pela topologia e características dos neurônios, onde as funções estão localizadas.

### 2.3 - *Autoencoder Variacional (Variational Autoencoder)*

O Autoencoder pode ser definido como um tipo de rede neural, sendo utilizado principalmente para tentar recriar e regenerar a entrada ao qual ele foi treinado. Para tentar dividir suas funcionalidades, variantes do Autoencoder foram criadas. Existem o *Sparse*, *Denoising*, *Contractive*, *Concrete* e o *Variational* (GOODFELLOW; BENGIO; COURVILLE, 2016). Todas essas subdivisões se focam na geração e representação de características que foram aprendidas, no entanto, *Sparse*, *Denoising*, *Contractive* e *Concrete* se focam na recriação de apenas alguns atributos para a saída, enquanto o *Variational* se foca em probabilidade para fazer a regeneração de um objeto aprendido.

O método de regeneração que o autoencoder propõe possui algumas aplicações práticas, como no reconhecimento facial, seleção de atributos em imagens, engenharia de linguagem natural, processamento digital de imagens e geração procedural de imagens.

### III. METODOLOGIA

O banco de dados *Respiratory Sound* foi originalmente criado para apoiar o desafio científico organizado pela ICBHI, esse banco de dados possui 920 áudios de 126 pacientes diferentes. Contudo, 793 amostras de sons respiratórios das 920 amostras da base de dados pertenciam à doença DPOC, mostrando a necessidade de balancear os dados.

Neste artigo foi analisado o uso prático de Autoencoder Variacional para ampliação e balanceamento dos dados com o objetivo de auxiliar o desenvolvimento de classificadores de sons respiratórios. A técnica utilizada visa aumentar os dados a partir de sons já processados anteriormente, os quais foram transformados em mel-espectrogramas.

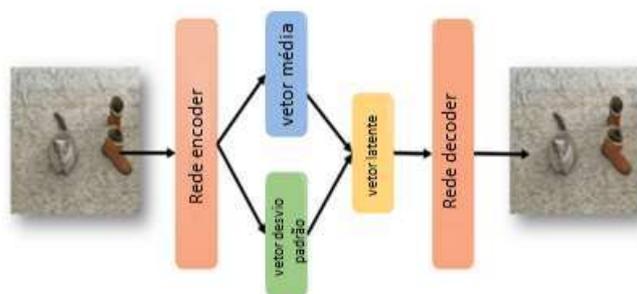
A base de dados possuía amostras de 7 doenças diferentes sendo elas Bronquiectasia, Bronquiolite, DPOC, Pneumonia, Infecção do Trato Respiratório Superior (URTI no inglês), Infecção do Trato Respiratório Inferior (LRTI no inglês) e Asma (ROCHA *et al.*, 2018). LRTI possui 2 amostras e Asma possui apenas 1 amostra. A pouca quantidade de amostras dessas doenças não garante uma boa representação do espaço amostral correspondente a cada uma delas, os dados gerados pelo autoencoder tenderiam a ser muito próximos dessas amostras. Optou-se então pela não utilização dessas doenças na pesquisa.

#### 3.1 – Arquitetura do Autoencoder Variacional

Autoencoders variacionais são modelos generativos profundos de redes neurais, ou seja, não só conseguem replicar a entrada, mas também gerar novos dados a partir do vetor do espaço latente. Esse espaço latente é regularizado no treinamento para garantir boas propriedades e tornar processos generativos possíveis. São formados por uma rede de codificação, ou encoder, e uma rede de decodificação, ou decoder (GOODFELLOW; BENGIO; COURVILLE, 2016).

A regularização do espaço latente tende a fazer as distribuições criadas pelo encoder próximas a uma distribuição normal padrão, por meio do desvio padrão e a média. A Figura 1 ilustra o esquema de um Autoencoder Variacional com as camadas de desvio padrão e média, os quais são usados para fazer o *sample* ou a amostragem do vetor latente.

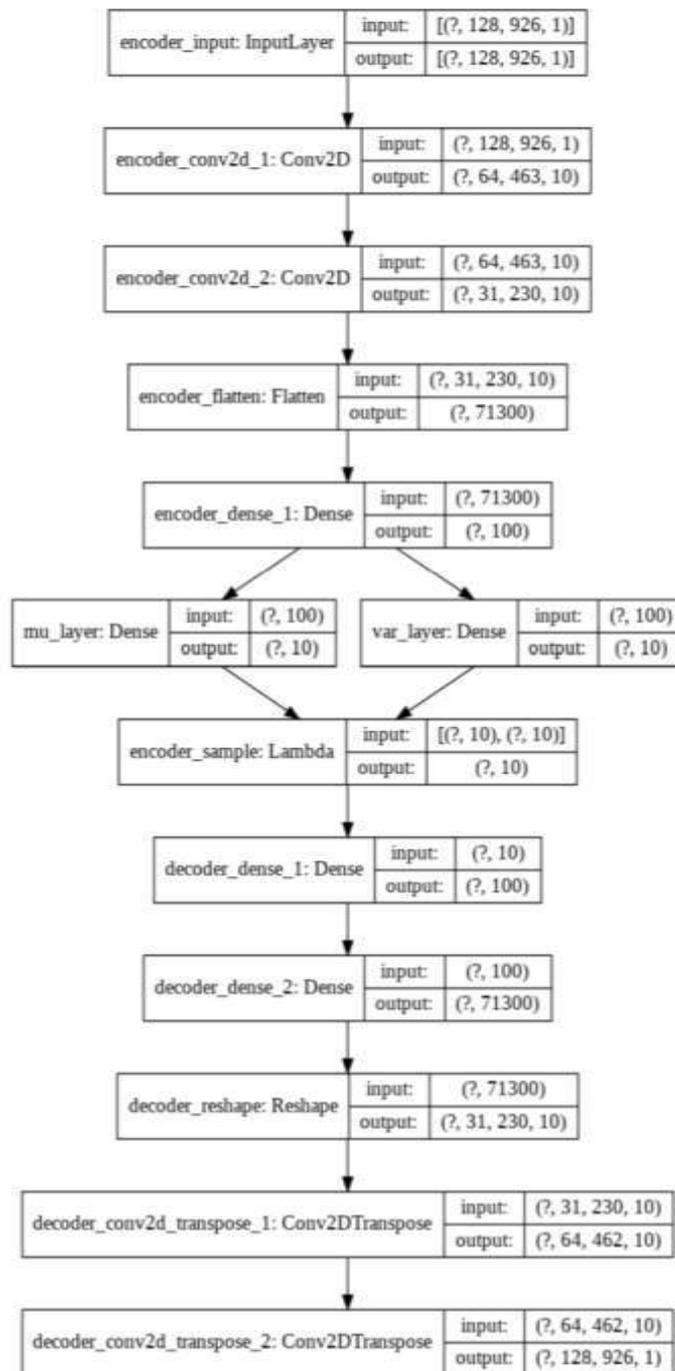
Figura 1 – Esquema de camadas do Autoencoder Variacional



Fonte: GARCIA-ÓRDAS, 2020.

A configuração do VAE criada para fazer o aumento de dados dos áudios respiratórios das classes correspondentes às doenças Bronquiectasia, Bronquiolite, URTI e Pneumonia, além dos áudios de pessoas saudáveis, é mostrada na Figura 2.

Figura 2 – Esquema da arquitetura do VAE



Fonte: GARCIA-ÓRDAS, 2020.

A partir dos modelos criados para encoder e decoder o VAE foi treinado separadamente para cada uma das 5 classes.

Um otimizador tem como finalidade diminuir o erro entre os resultados alcançados pela rede e os resultados desejáveis. Foi usado no treinamento o otimizador

Adam, com taxa de aprendizagem de 0,0005, o qual é um método de otimização que adapta a taxa de aprendizagem para cada amostra. O treinamento do autoencoder variacional foi feito com 100 épocas e tamanho de lote igual a 16.

### 3.2 – Espaço latente

O espaço latente é a camada oculta onde ocorre o redimensionamento dos dados, definindo o grau e qualidade da compressão (Data Science Academy, 2022). Como é necessário um modelo para reconstruir os dados compactados (decoder), ele deve aprender a armazenar todas as informações relevantes e desconsiderar o ruído. Essa é a importância da compressão, permitir que as informações irrelevantes sejam desconsideradas.

Para haver garantia que os dados sejam reconstruídos do espaço latente de forma mais próxima possível dos dados de entrada, neste caso mel-espectrogramas das amostras de sons pulmonares, deve-se primeiro visualizá-lo.

É possível obter essa visualização utilizando o encoder para gerar predições das amostras de áudios da classe treinada, nestas predições do encoder são gerados pontos com coordenadas x e y no espaço latente, em seguida é plotado um gráfico 2D com as predições no espaço latente. O gráfico mencionado será apresentado na seção Resultados.

Os algoritmos usados na pesquisa foram codificados na linguagem python que possui um conjunto de bibliotecas as quais auxiliam o desenvolvimento de redes neurais, por exemplo as bibliotecas *Keras* e *Tensorflow*, e a biblioteca para visualização de dados, *matplotlib*. O trecho do código responsável por traçar o gráfico do espaço latente é mostrado na Figura 3, onde é usada a biblioteca *matplotlib.pyplot* com as funções *figure*, *scatter*, *colorbar*, *xlabel* e *ylabel* para formação da figura, *show* para visualizar de fato a imagem e z é o espaço latente.

Figura 3 – Rotina responsável pela plotagem do espaço latente

```
def plot_label_clusters(vae, data):  
    # display a 2D plot of the classe in the latent space  
    z_mean, _, _ = vae.encoder.predict(data)  
    print(z_mean)  
    plt.figure(figsize=(12, 10))  
    plt.scatter(z_mean[:, 0], z_mean[:, 1])  
    plt.colorbar()  
    plt.xlabel("z[0]")  
    plt.ylabel("z[1]")  
    plt.show()  
  
plot_label_clusters(vae, data)
```

Fonte: Autoria própria.

### 3.3 – Geração de novos dados

Tendo conhecimento do espaço latente é possível escolher as coordenadas de um ponto dentro do mesmo e gerar um novo conjunto de dados com a predição do decoder. Selecionando intervalos no eixo x e no eixo y, escolheu-se vários pontos de uma vez. A partir desses pontos foram geradas predições no decoder, ou seja, foram gerados dados para novas imagens de mel-espectrogramas.

O código responsável por essa rotina pode ser visto na Figura 4, onde usou-se a biblioteca *numpy*, com a função *np.linspace* para separar *n* valores dentro dos intervalos de x e y escolhidos. A função *vae.decoder.predict* gera um novo conjunto de dados para

cada ponto de coordenadas  $x$  e  $y$ , no caso da rede esse ponto é uma amostra do espaço latente  $z$  representado pela variável  $z\_sample$ .

Figura 4 – Rotina responsável pela geração de novos dados

```
def plot_new_data_save(vae, x_min_scale, x_max_scale, y_min_scale, y_max_scale, n=4, figsize=10):  
  
    digit_size1 = 128  
    digit_size2 = 926  
  
    number_generated = 737  
    grid_x = np.linspace(x_min_scale, x_max_scale, n)  
    print(grid_x)  
    grid_y = np.linspace(y_min_scale, y_max_scale, n)  
    print(grid_y)  
    for i, yi in enumerate(grid_y):  
        for j, xi in enumerate(grid_x):  
            z_sample = np.array([[xi, yi]])  
            print(z_sample)  
            x_decoded = vae.decoder.predict(z_sample)  
            digit = x_decoded[0].reshape(digit_size1, digit_size2)
```

Fonte: Autoria própria.

## IV. RESULTADOS

A técnica explorada neste artigo visa obter novos conjuntos de dados os quais possam formar novos mel-espectrogramas. Os mel-espectrogramas são representações de áudio que utilizam a escala mel para serem formados. Essa escala é perceptiva de tons considerados iguais em distância um do outro pelos ouvintes, pois os humanos conseguem perceber melhor diferenças de frequências mais baixas.

Os resultados obtidos a partir das técnicas mencionadas nas seções anteriores envolvem as etapas de treinamento do VAE e visualização do espaço latente, bem como a obtenção de novos mel-espectrogramas referentes a cada uma das 5 classes treinadas.

### 4.1 – A arquitetura treinada

Testes realizados com espaço latente de dimensão superior a 2, tiveram resultados similares ou piores aos testes com espaço latente de 2 dimensões, podendo indicar sobre ajuste às amostras. A Figura 5 mostra as perdas na última época do treinamento. Então a arquitetura da rede foi modificada no espaço latente, o qual ficou apenas com 2 dimensões ao invés de 10 dimensões. Essa mudança tornou as informações ainda mais compactadas.

Figura 5 – Perdas do autoencoder variacional após treinamento. As perdas são, respectivamente, perda total (*loss*), perda de reconstrução (*reconstruction\_loss*) e divergência de Kullback-leibler (*kl\_loss*). Em (a) são mostradas as perdas para o espaço latente de 10 dimensões. Em (b) são mostradas as perdas para o espaço latente de 2 dimensões

```
(a)  
Epoch 100/100  
58/58 [=====] - 29s 493ms/step - loss: 28503.0243 - reconstruction_loss: 28611.8613 - kl_loss: 41.57  
56  
  
(b)  
Epoch 100/100  
58/58 [=====] - 27s 466ms/step - loss: 28464.4351 - reconstruction_loss: 28747.7383 - kl_loss: 13.08  
40
```

Fonte: Autoria própria.

A divergência de Kullback-leibler, uma medida da diferença entre duas distribuições de probabilidade (GOODFELLOW; BENGIO; COURVILLE, 2016), mostrou-se maior no treinamento com espaço latente de 10 dimensões.

Pode-se observar o arranjo da rede nas Figuras 6 e 7, correspondentes ao encoder e decoder, respectivamente. O encoder possui uma camada *input* para inserção dos dados, duas camadas de convolução, uma camada *flatten* onde os dados são convertidos em um único vetor, as camadas *dense* para condensação dos dados mais importantes e uma camada de *sampling*.

Figura 6 - Arquitetura do encoder do VAE

```

Model: "encoder"
-----
Layer (type)                Output Shape          Param #   Connected to
-----
encoder_input (InputLayer)  [(None, 128, 926, 1) 0
-----
encoder_conv_1 (Conv2D)     (None, 64, 463, 10)  50        encoder_input[0][0]
-----
encoder_conv_2 (Conv2D)     (None, 31, 230, 10)  1610     encoder_conv_1[0][0]
-----
flatten_2 (Flatten)         (None, 71300)        0         encoder_conv_2[0][0]
-----
encoder_dense (Dense)       (None, 100)          7130100  flatten_2[0][0]
-----
encoder_mu (Dense)          (None, 2)            202      encoder_dense[0][0]
-----
encoder_log_variance (Dense) (None, 2)            202      encoder_dense[0][0]
-----
sampling_2 (Sampling)       (None, 2)            0        encoder_mu[0][0]
                                     encoder_log_variance[0][0]
-----

```

Fonte: Autoria própria.

O decoder possui uma camada de *input*, duas camadas *dense*, nesse caso para descompactar os dados, uma camada *reshape* para retornar os dados ao tensor original, em seguida existem duas camadas de deconvolução.

Figura 7 - Arquitetura do decoder do VAE

```

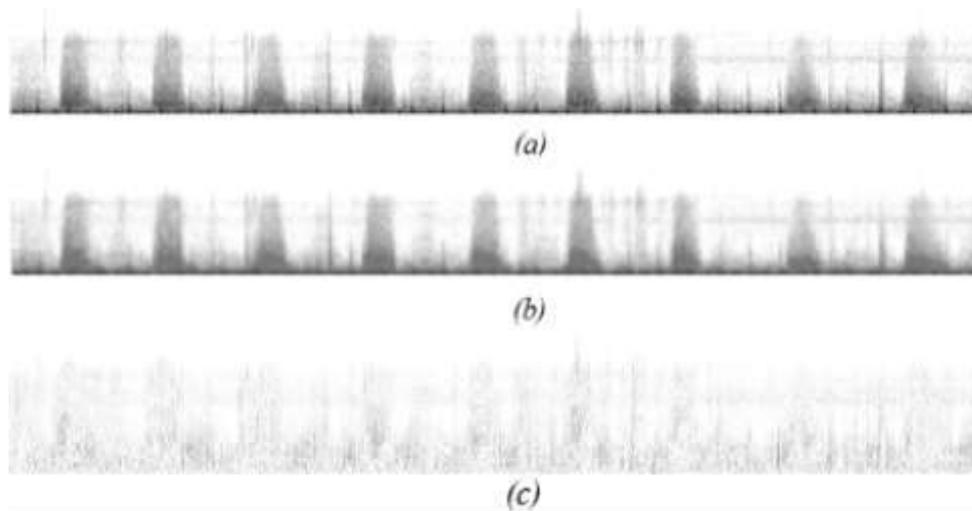
Model: "decoder_model"
-----
Layer (type)                Output Shape          Param #
-----
decoder_input (InputLayer)  [(None, 2)]          0
-----
decoder_dense_1 (Dense)     (None, 100)          300
-----
decoder_dense_2 (Dense)     (None, 71300)        7201300
-----
reshape_2 (Reshape)         (None, 31, 230, 10)  0
-----
decoder_conv_tran_1 (Conv2DT (None, 64, 462, 10)  1610
-----
decoder_conv_tran_4 (Conv2DT (None, 128, 926, 10)  810
-----
decoder_output (Conv2DTransp (None, 128, 926, 1)  11
-----

```

Fonte: Autoria própria.



Figura 9 – Mel-espectrogramas da classe Pneumonia. Em (a), mel- espectrograma usado no treinamento. Em (b), mel-espectrograma gerado. A imagem em (c) mostra a diferença de pixels entre o espectrograma gerado e o espectrograma original



Fonte: Autoria própria.

Figura 10 – Mel-espectrograma da classe Pneumonia gerado com espaço latente de 10 dimensões. Todos os espectrogramas com espaço latente desse tamanho, obtiveram resultados parecidos, não agregando as principais características dos espectrogramas da classe usados no treinamento



Fonte: Autoria própria.

## V. CONCLUSÃO

O presente trabalho propôs a utilização do VAE como ferramenta generativa de novos espectrogramas, para análise de sons respiratórios, com uso da linguagem de programação python, a fim de realizar o balanceamento de dados. A técnica possui uma série de etapas para proceder na análise dos dados a serem balanceados, incluindo o desenvolvimento da arquitetura e treinamento do VAE, análise do espaço latente e geração de novos espectrogramas a partir do espaço latente.

Através dos resultados alcançados, o estudo mostra uma alternativa para efetuar aumento e balanceamento de uma base de dados, auxiliando no desenvolvimento de pesquisas nas áreas de redes neurais e de classificadores, neste caso de sons respiratórios.

## VI. TRABALHOS FUTUROS

A partir dos resultados alcançados no balanceamento da base de dados, aumentando as amostras existentes das classes de doenças, pretende-se avançar nos estudos de redes neurais e inteligência artificial, especialmente para o desenvolvimento de um classificador de doenças pulmonares a partir dos dados gerados. O classificador poderia ser usado no desenvolvimento de sistemas de comparação de sons.

## VII. REFERÊNCIAS

- BZDOK, Danilo; ALTMAN, Naomi; KRZYWINSKI, Martin. **Statistics versus machine learning**. HHS Author Manuscripts, [s. l.], 3 abr. 2018. DOI 10.1038/nmeth.4642. Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6082636/>. Acesso em: 5 out. 2021.
- Data Science Academy. **Deep Learning Book**, 2022. Disponível em: <<https://www.deeplearningbook.com.br/variational-autoencoders-vaes-definicao-reducao-de-dimensionalidade-espaco-latente-e-regularizacao/>>. Acesso em: 10 jan. 2022.
- FÓRUM INTERNACIONAL DE SOCIEDADES RESPIRATÓRIAS. **O Impacto Global da Doença Respiratória**. [s. l.], ed. 2, 2017. Disponível em: [https://www.who.int/gard/publications/The\\_Global\\_Impact\\_of\\_Respiratory\\_Disease\\_POR.pdf](https://www.who.int/gard/publications/The_Global_Impact_of_Respiratory_Disease_POR.pdf). Acesso em: 5 out. 2021.
- GARCIA-ÓRDAS, M.T.; BENITEZ-ANDRADES, J.A.; CARGIA-RODRÍGUES, I.; BENAVIDES, C.; ALAIZ-MORETÓN, H. **Detecting respiratory pathologies using convolutional neural networks and variational autoencoders for unbalancing data**. *Sensors*. 2020;20:1214. doi: 10.3390/s20041214.
- GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron (2016). **Deep Learning**. MIT Press. ISBN 978-0262035613.
- LUGER, George F. **Inteligência Artificial**. 6. ed. [S. l.]: Pearson, 2013.
- MANDAL, Ananya. **História da doença pulmonar obstrutiva crônica**. *Medical Life Sciences News*, 26 fev. 2019. Disponível em: [https://www.news-medical.net/health/History-of-Chronic-Obstructive-Pulmonary-Disease\(Portuguese\).aspx](https://www.news-medical.net/health/History-of-Chronic-Obstructive-Pulmonary-Disease(Portuguese).aspx) Acesso em: 5 out. 2021.
- MARQUES, Alda; OLIVEIRA, Ana; JÁCOME, Cristina. Computerized Adventitious Respiratory Sounds as Outcome Measures for Respiratory Therapy: A Systematic Review. **Respiratory Care**, 2014, V. 59, N. 5. DOI: 10.4187/respcare.0276
- MCCULLOCH, Warren; PITTS, Walter. A Logical Calculus of Ideas Immanent in Nervous Activity. **Bulletin of Mathematical Biophysics**. 5 (4): 115–133 (1943). doi:10.1007/BF02478259.
- NORVIG, Peter Peter. **Inteligência Artificial**. 3. ed. [S. l.]: GEN LTC, 2013.
- RASCHKA, Sebastian. 2015. **Python Machine Learning**. Packt Publishing Ltd.
- ROCHA, B. M.; FILOS, D.; MENDES, L.; VOGIATZIS, I.; PERANTONI, E.; KAIMAKAMIS, E.; NATSIAVAS, P.; OLIVEIRA, A.; JACOME, C.; MARQUES, A.; PAIVA, R. P.; CHOUVARDA, I.; CARVALHO, P.; MAGLAVERAS, N. (2018). **A respiratory sound database for the development of automated classification**. pages 33–37. Springer.
- SIMON, Herbert A. **The Sciences of the Artificial**. 3. ed. [S. l.]: MIT Press, 1996. Disponível em: [https://monoskop.org/images/9/9c/Simon\\_Herbert\\_A\\_The\\_Sciences\\_of\\_the\\_Artificial\\_3rd\\_ed.pdf](https://monoskop.org/images/9/9c/Simon_Herbert_A_The_Sciences_of_the_Artificial_3rd_ed.pdf) . Acesso em: 5 out. 2021.
- THUN, M.J.; CARTER, B.D.; FESKANICH, D.; FREEDMAN, N.D.; PRENTICE, R.; LOPEZ, A.D.; HARTGE, P.; GAPSTUR, S.M. 50-Year Trends in Smoking-Related Mortality in the United States. **N. Engl. J. Med.** 2013, 368, 351–364.

VESTBO, Jørgen. **Global Strategy for the Diagnosis, Management, and Prevention of Chronic Obstructive Pulmonary Disease.** [S.l.]: Global Initiative for Chronic Obstructive Lung Disease. pp. 1–7, 2013.

World Health Organization. World health statistics 2017: monitoring health for the SDGs, Sustainable Development Goals. Geneva; 2017.

#### VIII. AGRADECIMENTOS

Este artigo é o resultado do projeto de PD&I IA\_Machine Learning for Pneumonia Diagnose, realizado pela instituição Universidade do Estado do Amazonas, em parceria com a Samsung Eletrônica da Amazônia Ltda., usando recursos da Lei Federal nº 8.387/1991, estando sua divulgação e publicidade em conformidade com o previsto no artigo 39.º do Decreto nº 10.521/2020.

Os autores agradecem a Universidade do Estado do Amazonas – Escola Superior de Tecnologia, a Agência de Inovação da UEA - AGIN e a Samsung Eletrônica da Amazônia Ltda. pelo apoio.

#### IX. COPYRIGHT

Direitos autorais: Os autores são os únicos responsáveis pelo material incluído no artigo.