

Redes Neurais Convolucionais (CNN) aplicadas na Classificação de ECG

Daniel Dantas do Amaral Ramos

Resumo— O sinal ECG mede a atividade elétrica do coração e pode ser usado como medida de monitoramento do sistema cardiovascular. Neste artigo, é avaliada uma técnica de classificação de sinais ECG (Eletrocardiograma) em 4 possíveis classes (Normal, Supraventricular ectopic beat, Ventricular ectopic beat e Fusion beat), utilizando redes neurais convolucionais (CNN) a partir de dados do banco MIT-BIH providos pela Physionet.

Abstract— The ECG signal measures the electrical activity of the heart and can be used as a monitoring measure for the cardiovascular system. In this paper, a technique for classifying ECG (Electrocardiogram) signals into 4 possible classes (Normal, Supraventricular ectopic beat, Ventricular ectopic beat, and Fusion beat) using convolutional neural networks (CNN) from MIT-BIH database data provided by Physionet is evaluated.

I. INTRODUÇÃO

O CORAÇÃO é um músculo que contrai de maneira rítmica, bombeando sangue para o corpo humano, dessas contrações se geram diferenciais elétricos em pontos específicos do corpo, que seguem um padrão, estes sinais podem ser capturados por equipamentos e usados para fins médicos [1]. O uso do sinal ECG (Eletrocardiograma) possibilita o monitoramento e gravação da atividade durante a operação do coração e é altamente usado por médicos cardiologistas para monitorar e diagnosticar a saúde do indivíduo [2].

Doenças cardiovasculares são uma grande causa de mortes humanas, responsáveis por 31% das mortes globais em 2016, das quais 85% aconteceram de ataque cardíaco [3]. Os paradigmas tradicionais de diagnóstico são baseados no histórico médico do paciente e exames clínicos, tais resultados são interpretados de acordo com o conjunto de parâmetros médicos quantitativos e usados para classificar as doenças.

Em vários casos, os paradigmas tradicionais baseados em regras para o diagnóstico não são suficientes devido à alta heterogeneidade dos dados e requerem grande esforço médico de análise e tempo para atingir acurácias apropriadas. Esta dificuldade é ainda mais evidente em localidades onde há falta de equipamentos e profissionais experientes, o que motiva a criação de uma técnica confiável e automática para o monitoramento de doenças do coração [4]. Nesse cenário, várias técnicas de diagnóstico auxiliado por computadores foram desenvolvidas, dentre as quais a utilização de inteligência artificial, mais especificamente *Deep Learning*, que se mostrou eficientemente acurado na classificação em um grande número de tarefas médicas [4]. Para a problemática de classificação de arritmia

em ECG, várias técnicas de *machine learning* e *Deep learning* são utilizadas, como Support Vector Machines, multi-layer perceptron, decision trees, etc [2].

Todavia, o uso de sistemas apoiados por computadores na tarefa de classificação de arritmia enfrenta grandes desafios, como:

1. Os sintomas de arritmia podem não ser vistos durante a captura do sinal ECG;
2. As propriedades do sinal ECG, como período e amplitude, variam de pessoa para pessoa e dependem de fatores como idade, gênero, condição física, dentre outros;
3. A morfologia do sinal ECG é frequentemente não estacionária;
4. O volume de dados a ser considerado é grande;
5. Interferências, ruídos e artefatos podem resultar em variações morfológicas e discrepâncias no sinal ECG capturado.

Este artigo, portanto, apresenta uma proposta de modelo de machine learning com redes neurais convolucionais (CNN) para classificação de tipos de arritmia em sinais ECG. Na Seção II é feita uma breve fundamentação teórica, na Seção III é demonstrado a metodologia adotada neste trabalho. Na Seção IV são apresentados os resultados alcançados com a rede treinada. Por fim, na Seção V são feitas as conclusões acerca dos resultados e da problemática deste artigo.

II. FUNDAMENTAÇÃO TEÓRICA

A. Classificação de Arritmia em ECG

A morfologia do ECG reflete o estado de saúde do coração. Ao medir intervalos de tempo no ECG, um médico cardiologista pode determinar quanto tempo a onda elétrica leva para passar pelo sistema de condução elétrica do coração. Esta informação ajuda a descobrir se a atividade elétrica é regular ou irregular, rápida ou lenta. Segundo [4], ao medir a força da atividade elétrica, um cardiologista é capaz de descobrir se partes do coração são muito grandes ou se estão sobrecarregadas de trabalho, além de outras possibilidades de diagnóstico providas pelo ECG.

Uma batida de sinais de ECG pode ser observada por cinco ondas características: P, Q, R, S, e T, como mostrado na Figura 1. Onda P significa despolarização atrial; o complexo QRS corresponde a despolarização dos ventrículos direito e esquerdo, e contração dos músculos ventriculares; onda T representa a repolarização dos ventrículos [5].

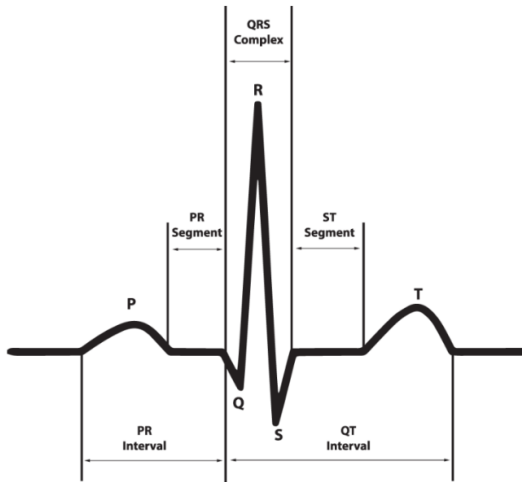


Figura 1 – Sinal ECG. Fonte: [5]

Qualquer desordem característica na atividade elétrica do coração afeta o sinal ECG, batizando a característica da arritmia. Tais características podem ser diagnosticadas pelo ECG, levando a resultados de diagnóstico como Fibrilação Atrial, Contração Atrial Prematura e Contração Ventricular Prematura, Infarto do miocárdio, dentre outros tipos de arritmias [4].

A anormalidade do sinal de ECG é geralmente identificada pelo uso de técnicas de classificação [5], que são alcançados através da extração das características do ECG que podem discriminar as categorias definidas [6]. Convencionalmente, as características são derivadas da magnitude, duração e área do QRS e T [4].

Para o bom entendimento do problema e resultados deste trabalho, os itens apresentados acima resumem a importância do sinal ECG e dão noção de como este sinal é usado na prática para auxiliar no diagnóstico.

B. Métricas de Avaliação de Performance

Adentrando o tópico técnico de redes neurais, este item apresenta as métricas quantitativas mais comuns para avaliação de performance de classificação. O resultado de uma classificação advinda de um modelo treinado pode ser negativo ou positivo, sendo que os resultados positivos podem ser multi classes, a depender do banco de dados utilizado e classes presentes. Então, podemos sumarizar o resultado da classificação nestes estados:

- VP significa verdadeiro positivo, indica que a amostra foi classificada corretamente como sendo de uma determinada classe.
- VN significa verdadeiro negativo, indica que a amostra foi classificada corretamente como não sendo de uma determinada classe.
- FP significa falso positivo, indica que a amostra foi classificada erroneamente como de um determinado tipo de classe.
- FN significa falso negativo, indica que a amostra foi classificada erroneamente que não pertencia a uma classe, quando na realidade pertencia.

A métrica acurácia é definida como:

$$\frac{(VP+VN)}{(VP+VN+FP+FN)} \quad (1)$$

A acurácia é uma das métricas mais utilizadas, já que divide o total de acertos pelo total de valores presentes na base de dados, e diz quanto o modelo acertou das classificações possíveis, então sua interpretação é a mais intuitiva.

A métrica precisão é definida por:

$$\frac{VP}{(VP+FP)} \quad (2)$$

Esta métrica responde a seguinte pergunta: Qual a proporção de identificações positivas foi realmente correta? No caso de diagnóstico de faltas, a métrica indica quantos casos previstos como determinada falha, realmente eram daquela falha.

A matriz de confusão é uma ferramenta gráfica importante pois permite analisar rapidamente o desempenho do sistema, os valores que compõem a matriz advém da comparação entre o conjunto de teste e a predição do sistema, que computa todos os acertos e erros.

A métrica “AUC” (*Area under the curve*), ou “área sob a curva” em tradução literal, é uma métrica utilizada para avaliar a performance em modelos de classificação. A intuição para esta métrica é a seguinte: “É a probabilidade que o modelo tem de distinguir entre as classes”.

C. Redes Neurais Convolucionais

Uma rede neural convolucional consiste em três camadas: camada de entrada, camada oculta e camada de saída, que é semelhante às redes neurais tradicionais como as redes neurais artificiais (RNA). A diferença entre CNN e RNA é apresentada na camada oculta. A camada oculta de um modelo CNN é geralmente composta de três camadas, ou seja, camada de convolução, camada de pooling e camada totalmente conectada [7].

As camadas convolucionais aplicam uma forma de convolução discreta entre um tensor de entrada, como uma imagem com canais diferentes, e uma função representada por uma matriz 2-D menor chamada filtro ou *kernel* (núcleo), como ilustrado na figura 2. A declaração matemática do processo de convolução bidimensional entre duas funções f e g é apresentado a seguir:

$$(f * g)(x, y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} g(i, j) f(x - i, y - j) \quad (3)$$

onde x e y são as duas variáveis de ambos f e g , por exemplo uma linha e coluna de um pixel de uma imagem, e i e j são os valores de indexação que passam de f para g [8].

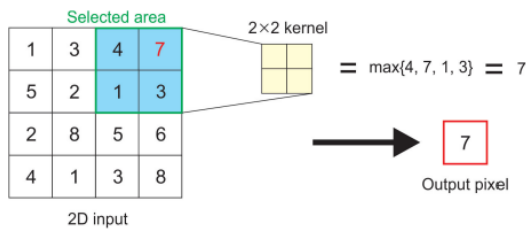


Figura 2 – Etapa de convolução e Max pooling. Fonte: [8]

A figura 2 exemplifica o processo de convolução de uma entrada com duas dimensões. Para a realidade do caso aplicado neste trabalho, o processo de convolução se deu em um cenário cuja entrada era uma informação em uma dimensão apenas, especificamente, os valores da série temporal do ECG. O processo de convolução ocorreu da mesma maneira onde o *kernel* foi um vetor de 2 valores.

Uma camada convolucional é frequentemente seguida por uma camada de normalização dos lotes, ou *batch normalization* [9], que é uma camada proposta em 2015 com a finalidade de realizar uma normalização nos dados de saída da camada anterior, garantindo que estes tenham uma média 0 e um desvio padrão de 1.

A saída de uma camada convolucional é frequentemente chamada de mapa de características (*feature map*) ou mapa de ativação, uma vez que se espera que carregue as características extraídas dos dados.

A camada de pooling, também conhecida como camada de sub-amostragem, reduz a resolução espacial de sua entrada. Uma camada de pooling consiste em um núcleo vazio com um valor de passada (*stride*). Os métodos de agrupamento mais populares são o agrupamento máximo (*maxpooling*), que extrai o maior valor percebido pelo kernel e o envia ao mapa de características [8].

Frequentemente, as arquiteturas de redes neurais convolucionais contam com a presença de uma camada totalmente conectada (*Fully connected*) na saída para servir como classificador, esta camada consiste de uma rede feedforward tradicional e conta com a presença, geralmente, de uma função de ativação do tipo softmax na saída para realizar classificação multi-classe, que é o caso deste estudo [8].

III. METODOLOGIA

Neste artigo, foi proposto como estudo de caso a utilização de redes neurais convolucionais para classificação de 4 tipos diferentes de classes relacionadas à arritmias cardíacas, são elas: Normal, *Supraventricular ectopic beat*, *Ventricular ectopic beat* e *Fusion Beat*.

A. Base de Dados

Os dados utilizados são provenientes do repositório MIT-BIH (The Massachusetts Institute of Technology - Beth Israel Hospital). Este banco de dados contém 48 gravações de batimentos cardíacos numa taxa de frequência de 360 Hz por aproximadamente 30 minutos de 47 indivíduos diferentes. Cada gravação contém 2

faixas de sinais ECG, sendo a principal delas a faixa A, pois esta contém o complexo QRS de maneira mais proeminente. A figura 3 mostra um exemplo de sinal ECG proveniente do banco.

Cada batimento foi anotado por pelo menos 2 cardiologista, essas anotações foram utilizadas para realizar a classificação em cada uma das 4 classes.

Os dados adotados neste trabalho foram tratados com a segmentação somente do complexo QRS e sua respectiva anotação, como exemplificado na figura 4. Contendo ao total 100.693 amostras de dados, com 89.857 amostras do tipo normal, 3.026 amostras do tipo *Supraventricular ectopic beat*, 7.008 amostras do tipo *Ventricular ectopic beat* e 802 amostras do tipo *Fusion beat*.

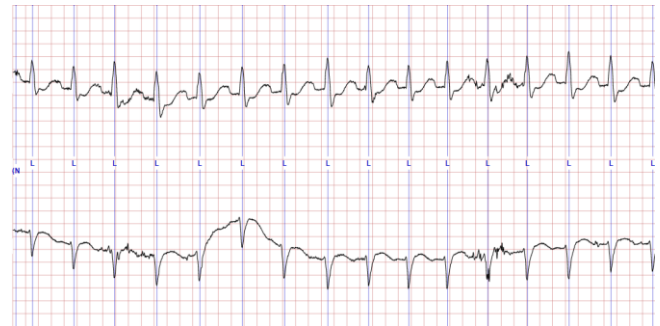


Figura 3 – Sinal ECG da base MIT-BIH. Fonte: physionet.org

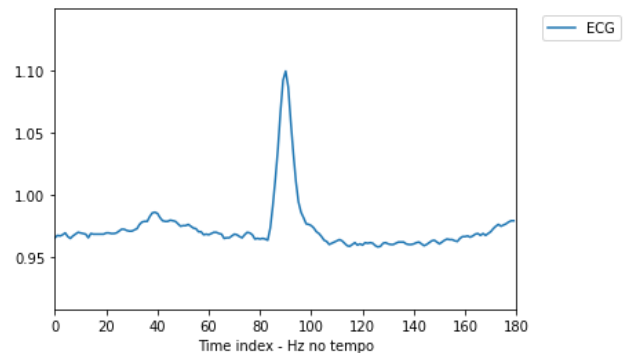


Figura 4 – Complexo QRS segmentado do tipo “Normal”.
Fonte: Autor

Para realizar o treinamento e teste, preservando a boa capacidade de generalização da rede, os dados foram divididos em 2 conjuntos de dados (*datasets*), divididos baseados no critério de pacientes únicos, ou seja, um mesmo paciente não podia pertencer a ambos os conjuntos, justamente para não comprometer a capacidade de generalização da rede ao fazer a predição em dados com características morfológicas que a rede usou para o treinamento.

Assim, foi adotada a proposta de divisão de [1], O primeiro conjunto é composto por todos os batimentos cardíacos presentes nas gravações: 101, 106, 108, 109, 112, 114, 115, 115, 116, 118, 119, 122, 124, 201, 203, 205, 207, 208, 209, 215, 220, 223 e 230, chamado de Dataset 1 (DS1). Enquanto o segundo é composto de todos os batimentos cardíacos presentes nas gravações: 100, 103, 105, 11, 113, 117, 121, 123, 200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233 e 234, chamado de Dataset 2 (DS2).

Neste trabalho, foi utilizado o DS1 para construir o modelo de classificação, enquanto a DS2 foi reservada para o teste.

B. Construção e Treinamento da Rede Neural

A arquitetura de rede escolhida segue o padrão das últimas arquiteturas apresentadas em literatura, com a soma de camadas convolucionais, camadas de “batch normalization”, camadas de “pooling”, e camadas de “feedforward” para fazer a classificação com função softmax. A tabela 1 abaixo mostra as camadas utilizadas neste estudo.

TABELA I
CAMADAS IMPLEMENTADAS

Camada	Tipo	Número de filtros	Tamanho do Kernel	Stride (Passo)
Camada 1	Conv1D	64	6	1
Camada 2	Batch Normalization	-	-	-
Camada 3	Max Polling	-	3	2
Camada 4	Conv1D	64	3	1
Camada 5	Batch Normalization	-	-	-
Camada 6	Max Polling	-	2	2
Camada 7	Conv1D	64	3	1
Camada 8	Batch Normalization	-	-	-
Camada 9	Max Polling	-	2	2
Camada 10	Flatten	-	-	-
Camada 11	Feed Foward	64	-	-
Camada 12	Feed Foward	32	-	-
Camada 13	Feed Foward	4	-	-

A abordagem de convolução em uma dimensão foi escolhida tendo-se como objetivo alinhar performance com custo de processamento, já que com uma dimensão não estamos trabalhando com imagens, o que torna o processo de tratamento dos dados e treinamento da rede menos pesado.

O modelo foi criado, treinado e avaliado utilizando-se softwares open-source escritos na linguagem de programação “Python”. Foram utilizadas, especificamente, as seguintes bibliotecas: Pandas e Numpy para tratamento e modelagem de dados, Matplotlib e Seaborn para visualização de gráficos, Tensorflow e Keras para criação, treinamento e avaliação do modelo de aprendizagem de máquina. O código foi escrito utilizando-se o recurso “Google Colaboratory”, que é um serviço de máquinas virtuais gratuito para processamento de código python em “Notebook”.

A arquitetura adotada receberá em sua entrada um vetor correspondente ao sinal segmentado do complexo QRS de dimensão (180,1), ou seja, 180 valores correspondentes à amplitude do sinal elétrico do complexo QRS que irão passar pela convolução.

Já os valores de anotação, ou seja, as classes as quais cada complexo QRS pertencem tanto para o treinamento quanto para o teste, foram tratados a fim de representarem as classes no formato categórico, ou seja, correspondente à função Softmax de saída. A

tabela 2 exemplifica este caso.

TABELA II
CLASSES NO FORMATO CATEGÓRICO

Classe	Codificação
Normal – “N”	[1,0,0,0]
<i>Supraventricular ectopic beat</i> – “S”	[0,1,0,0]
<i>Ventricular ectopic beat</i> – “V”	[0,0,1,0]
<i>Fusion Beat</i> – “F”	[0,0,0,1]

Para o treinamento, foi adotado 40 como número de épocas, todas as camadas da arquitetura tem como função de ativação a “Relu”, somente a última camada (feed foward) que possui a função de ativação “Softmax” para realizar a classificação.

O modelo criado com a biblioteca Keras possui algumas configurações úteis para testes e tratamento dos resultados. O item configurável “EarlyStopping” para o treinamento quando uma métrica monitorada chega em um gatilho pré-configurado. Neste trabalho, foi adotado a parada de treinamento caso o valor da função custo de validação pare de decrescer durante 8 épocas.

No caso deste estudo, foi usado para configurar o valor do “Loss”, que é a função custo que queremos minimizar, a função “categorical_crossentropy”, que é a função recomendada pela documentação oficial do Keras quando quer-se fazer a classificação de multi-classes, sendo estas no modelo de representação “one-hot category”, ou seja, categorizadas como apresentado na tabela 2.

O otimizador usado foi o “Adam”, proposto e 2015 por [10], este otimizador, que tem a função de atualizar os pesos de treinamento, é uma extensão do clássico “gradiente descendente estocástico”. Este otimizador atualiza o passo de aprendizagem de acordo com os dados de treinamento, iterativamente.

IV. RESULTADOS

Após o treinamento das arquiteturas de rede, utilizando os critérios de acurácia na base de validação, foi selecionada a arquitetura de 64 filtros nas 3 camadas convolucionais, com o critério de parada de 8 épocas após a função custo parar de decrescer na base de validação. Para o teste, foi utilizado o Dataset 2, contendo amostras que a rede treinada nunca tinha visto antes. A tabela 3 exemplifica os principais testes feitos com configurações diferentes, a tabela 4 apresenta os resultados gerais do modelo. A figura 5 mostra a matriz de confusão obtida e as figuras 6 e 7 mostram as curvas de acurácia função custo durante o processo de treinamento.

TABELA III
MÉTRICAS DE TESTES

Número de Épocas	Número de filtros	Acurácia	Precisão
40	Camada 1:64 Camada 4:128 Camada 7:256	0.926	0.926
40	Camada 1:64 Camada 4:128 Camada 7:128	0.847	0.847
50	Camada 1:64 Camada 4:64 Camada 7:64	0.917	0.917
40	Camada 1:64 Camada 4:64 Camada 7:64	0.936	0.936

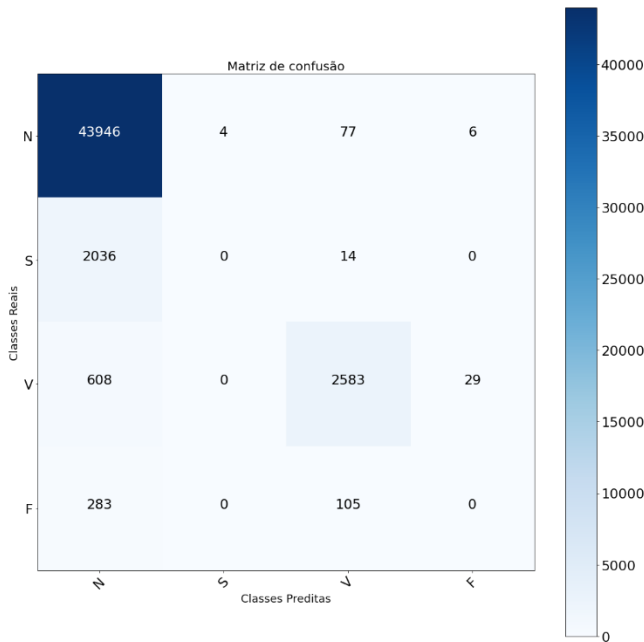


Figura 5 – Matriz de confusão. Fonte: Autor

TABELA IV
MÉTRICAS GERAIS

Número de Épocas	Número de filtros	Acurácia	Precisão	AUC
40	Camada 1:64 Camada 4:64 Camada 7:64	0.936	0.936	0.660

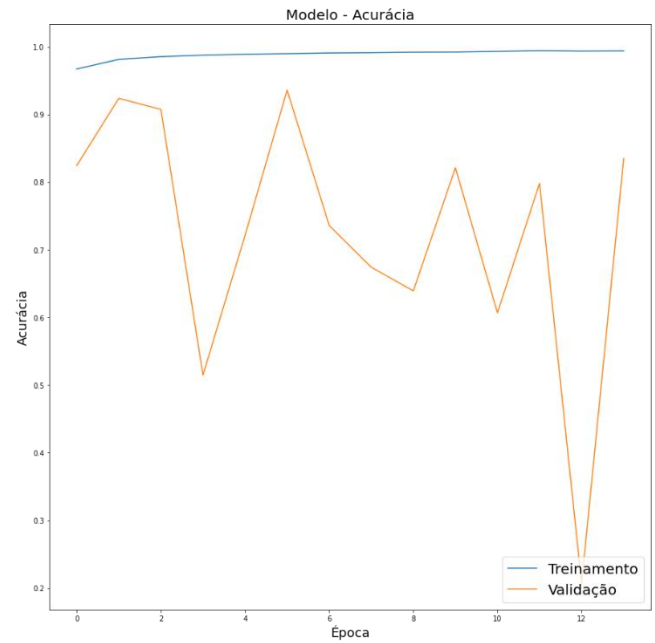


Figura 6 – Acurácia durante as épocas. Fonte: Autor

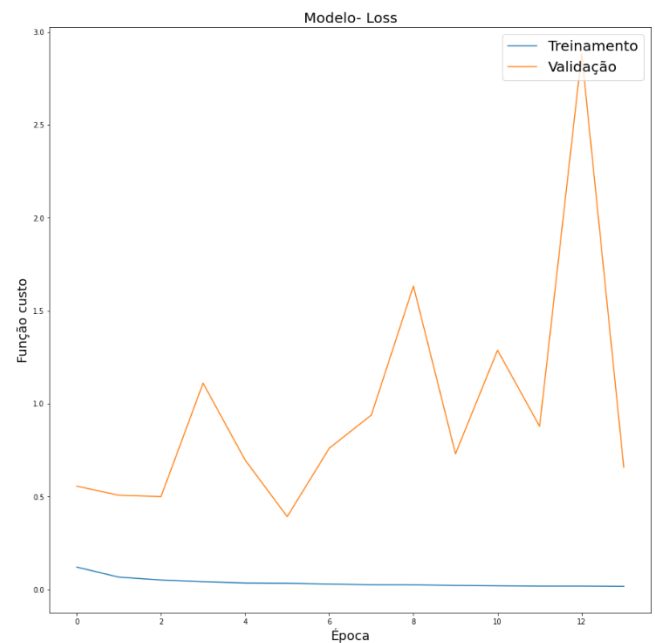


Figura 7 – Acurácia durante as épocas. Fonte: Autor

A. Resultados do modelo

Podemos perceber que os resultados do modelo, em termos percentuais, apresentam bons números com acurácia de 93%. Todavia, com análises mais completas e aplicadas ao contexto (Classificação de arritmia em ECG), notamos que o modelo criado precisa de melhorias pois quando se trata de amostras do tipo “S”, o modelo errou todas as predições, indicando que estas amostras pertenciam, na realidade, à classe normal (“N”). Além disso, podemos observar na métrica “AUC” que a capacidade de discernimento dentre classes do nosso modelo está baixa, somente 66%. Todos esses fatos podem ser originários ao fato de que os

dados em nosso banco são desbalanceados, com um grande número de dados referente à classe “N”.

Segundo [11], quando existe desequilíbrio de classe dentro dos dados de treinamento, os modelos tipicamente classificarão em excesso o grupo majoritário devido a sua maior probabilidade anterior. E como resultado, as amostras pertencentes aos grupos minoritários são classificadas erroneamente.

B. Lidando com dados não balanceados

Os métodos para lidar com o desequilíbrio de classe na aprendizagem de máquinas podem ser agrupados em três categorias: técnicas de nível de dados, métodos de nível de algoritmo e abordagens híbridas [11]. As técnicas a nível de dados tentam reduzir o nível de desequilíbrio através de vários métodos de amostragem de dados. Os métodos a nível de algoritmo, incluem a modificação o momento da aprendizagem, já os sistemas híbridos combinam estrategicamente tanto métodos de amostragem quanto algoritmos [11].

Neste trabalho, serão adotadas técnicas a nível de dados, especificamente amostragem para menos e para mais (“under-sampling” e “over-sampling”). De maneira simples, a reamostragem aleatória para menos, ou subamostragem descarta amostras aleatórias da maioria enquanto a reamostragem aleatória para mais, ou sobreamostragem, duplica as amostras aleatórias do grupo minoritário [12].

As técnicas só serão adotadas no Dataset 1(dados de treinamento) e a tabela 5 resume o procedimento adotado.

TABELA V
REMOSTRAGEM DOS DADOS

Classe	Quantidade após reamostragem
Normal – “N”	20000
<i>Supraventricular ectopic beat</i> – “S”	20000
<i>Ventricular ectopic beat</i> – “V”	20000
<i>Fusion Beat</i> – “F”	20000

A matriz de confusão para os dados balanceados pode ser vista na figura 8. A tabela 6 mostra os testes performados com o novo conjunto de dados.

TABELA VI
MÉTRICAS DE TESTES

Número de Épocas	Número de filtros	Acurácia	Precisão	AUC
40	Camada 1:64 Camada 4:64 Camada 7:64	0.821	0.821	0.706

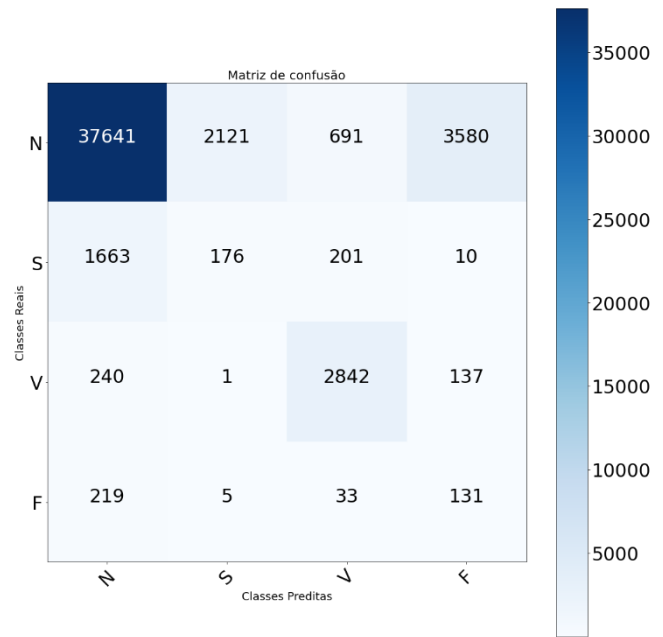


Figura 8 – Matriz de confusão. Fonte: Autor

Observa-se pela matriz de confusão que o modelo com dados balanceados continua tendendo à classe “N”, talvez por algum motivo morfológico, todavia os resultados de previsão para as outras classes performaram melhor do que para o caso de dados não balanceados, demonstrando que o balanceamento melhorou a classificação de múltiplas classes nesse caso.

Destaque para as classes “F” e “S”, para as quais o modelo não tinha acertado nenhuma previsão e agora acertou 9% para a classe “S” e 34% para a classe “F”. As classes que continuam com as melhores porcentagens de previsão acertada são a classe “N”, com 85% de acerto e a classe “V” com 88% de acerto. Outra métrica que comprova a melhora após o balanceamento é a AUC, que agora nos fala que o modelo tem 60% de chance de acertar uma previsão para a classe correta.

V. CONCLUSÕES

Este trabalho desenvolveu um modelo de rede neural utilizando redes convolucionais 1D para a classificação de arritmia em sinais ECG. Os resultados se mostraram pouco confiáveis para o uso comercial, todavia contribuem para avaliação e aprimoramento de técnicas de machine learning hoje utilizadas, em especial aquelas que tratam de bancos de dados não balanceados e que foram testadas neste trabalho.

Ficou evidente que a técnica de subamostragem e a técnica de sobreamostragem trazem resultados tímidos para o problema de classificação de arritmia, na arquitetura adotada, talvez por algum fator morfológico ou limitação de processamento da rede, fator que deve ser levado em consideração na concepção de novas propostas de arquitetura.

Como trabalhos futuros, almeja-se investigar novos modelos que se adequem melhor ao caso de ECG, em especial ao banco de dados disponível e usado neste trabalho, “MIT-BIH”, bem como modelos e técnicas que lidem melhor com o problema de dados não balanceados, tais como SMOTE, técnicas de machine learning e modelos híbridos.

REFERÊNCIAS

- [1] E. J. da S. Luz, W. R. Schwartz, G. Cámara-Chávez, e D. Menotti, “ECG-based heartbeat classification for arrhythmia detection: A survey”, *Computer Methods and Programs in Biomedicine*, vol. 127, p. 144–164, abr. 2016, doi: 10.1016/j.cmpb.2015.12.008.
- [2] M. Kachuee, S. Fazeli, e M. Sarrafzadeh, “ECG Heartbeat Classification: A Deep Transferable Representation”, em *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, jun. 2018, p. 443–444. doi: 10.1109/ICHI.2018.00092.
- [3] “Heart Disease and Stroke Statistics—2018 Update: A Report From the American Heart Association”. <https://www.ahajournals.org/doi/epub/10.1161/CIR.0000000000000558> (acessado 4 de julho de 2022).
- [4] Z. Ebrahimi, M. Loni, M. Daneshlab, e A. Gharehbaghi, “A review on deep learning methods for ECG arrhythmia classification”, *Expert Systems with Applications: X*, vol. 7, p. 100033, set. 2020, doi: 10.1016/j.eswx.2020.100033.
- [5] C.-H. Hsieh, Y.-S. Li, B.-J. Hwang, e C.-H. Hsiao, “Detection of Atrial Fibrillation Using 1D Convolutional Neural Network”, *Sensors (Basel)*, vol. 20, n° 7, p. E2136, abr. 2020, doi: 10.3390/s20072136.
- [6] X. Tang e L. Shu, “Classification of Electrocardiogram Signals with RS and Quantum Neural Networks”, *IJMUE*, vol. 9, n° 2, p. 363–372, fev. 2014, doi: 10.14257/ijmue.2014.9.2.37.
- [7] “Convolutional neural networks for medical image analysis: State-of-the-art, comparisons, improvement and perspectives”, *Neurocomputing*, vol. 444, p. 92–110, jul. 2021, doi: 10.1016/j.neucom.2020.04.157.
- [8] G.-A. Vargas-Hákim, E. Mezura-Montes, e H.-G. Acosta-Mesa, “A Review on Convolutional Neural Network Encodings for Neuroevolution”, *IEEE Transactions on Evolutionary Computation*, vol. 26, n° 1, p. 12–27, fev. 2022, doi: 10.1109/TEVC.2021.3088631.
- [9] S. Ioffe e C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, fev. 2015, doi: 10.48550/arXiv.1502.03167.
- [10] D. P. Kingma e J. Ba, “Adam: A Method for Stochastic Optimization”. arXiv, 29 de janeiro de 2017. doi: 10.48550/arXiv.1412.6980.
- [11] J. M. Johnson e T. M. Khoshgoftaar, “Survey on deep learning with class imbalance”, *J Big Data*, vol. 6, n° 1, p. 27, dez. 2019, doi: 10.1186/s40537-019-0192-5.
- [12] J. Van Hulse, T. M. Khoshgoftaar, e A. Napolitano, “Experimental perspectives on learning from imbalanced data”, em *Proceedings of the 24th international conference on Machine learning*, New York, NY, USA, jun. 2007, p. 935–942. doi: 10.1145/1273496.1273614.