

# A Competitive Structure of Convolutional Autoencoder Networks for Electrocardiogram Signals Classification

Alexandre Farias Baia<sup>1</sup>, Adriana Rosa Garcez Castro<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal do Pará (UFPA) – Belém, PA – Brazil

alexfarb@gmail.com, adcastro@ufpa.br

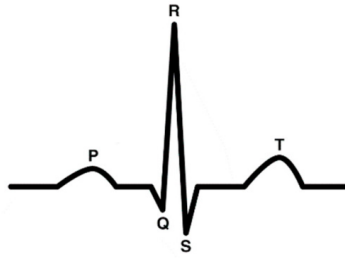
**Abstract.** *This paper presents the proposal of an electrocardiogram (ECG) signals classification system through a competitive structure of Convolutional Autoencoders (CAE). Two Convolutional Autoencoders were trained to reconstruct ECG signals for the cases of patients with arrhythmia and patients with signals considered normals. After the training, the two networks were arranged in a competitive parallel structure to classify these signals. For the development and testing of the system, the MIT-BIH Arrhythmia Database of ECG signals was used. An accuracy of 88,9% was achieved considering the database used for system testing.*

**Resumo.** *Este artigo apresenta a proposta de um sistema para classificação de sinais de eletrocardiograma (ECG) através de uma estrutura competitiva de Redes Neurais Convolucionais Auto-associativas. Duas redes neurais convolucionais auto-associativas foram treinadas para reconstrução de sinais ECG para os casos de sinais de pacientes com arritmia e pacientes com sinais considerados normais. Após o treinamento, as duas redes foram arranjadas em uma estrutura paralela competitiva para classificação dos Sinais ECG. Para desenvolvimento e teste do sistema foi utilizada a base de dados MIT-BIH Arrhythmia de sinais ECG. Uma acurácia de 88,9% foi alcançada considerando a base de dados usadas para teste do sistema.*

## 1. Introdução

O eletrocardiograma (ECG) é um método de diagnóstico não-invasivo que permite a obtenção de informações a respeito do ritmo cardíaco de um paciente. Através do ECG, anomalias no aparelho cardíaco, tais como arritmias, podem ser detectadas através da análise de variações da corrente cardíaca em função do tempo. Para a coleta dos sinais cardíacos, aparelhos de medição são colocados no corpo do paciente, onde para cada ponto de coleta pode-se obter um diagnóstico específico. A Figura 1 apresenta o aspecto que pode ser obtido através de um eletrocardiograma de um batimento cardíaco de um paciente com uma frequência normal. A Figura 1 também descreve as respectivas unidades de composição do sinal cardíaco [ECG 2018].

Para auxílio à detecção de anomalias em batimentos cardíacos e apoio à decisão médica, alguns estudos têm sido realizados e apresentados na literatura utilizando sinais ECG aliados as técnicas de Aprendizado de Máquina, tais como Redes Neurais Artificiais [Lassoued e Ketata 2018, Kelwade e Salankar 2015], Lógica difusa [Naik e Reddy 2016, Tandale et al. 2017], SVM [Ye et al. 2012], Computação por reservatório (RC) [Escalona-Moran 2015], entre outras.



**Figura 1. Ritmo cardíaco normal. Com a presença da onda P (despolarização do átrio), complexo QRS (despolarização dos ventrículos) e onda T (repolarização dos ventrículos). Fonte: [Isin e Ozdalili 2017].**

Considerando as Redes Neurais Artificiais, atualmente, as Redes Neurais baseadas em aprendizado profundo (*Deep Learning*) vêm se destacando cada vez mais na literatura, em diversas áreas de aplicação, principalmente nas áreas voltadas para reconhecimento e classificação de imagens. O aprendizado profundo permite que modelos computacionais aprendam representações de dados com múltiplos níveis de abstração [Lecun et al. 2015]. Além dos resultados promissores já alcançados para problemas envolvendo a aplicação voltada para reconhecimento de imagem representadas através de dados 2-D [Nardelli et al. 2018], algumas pesquisas vêm sendo desenvolvidas com aplicações em problemas de classificação com o uso de dados 1-D [Penha e Castro 2017], tais como dados de séries temporais. Dentre as redes neurais profundas destacam-se as *Long Short Term Memory* (LSTM), as Redes Neurais Auto-associativas, as Redes Auto-associativas Empilhadas (*Stacked autoencoder*) e as Redes Neurais Convolucionais (*Convolutional Neural Network*).

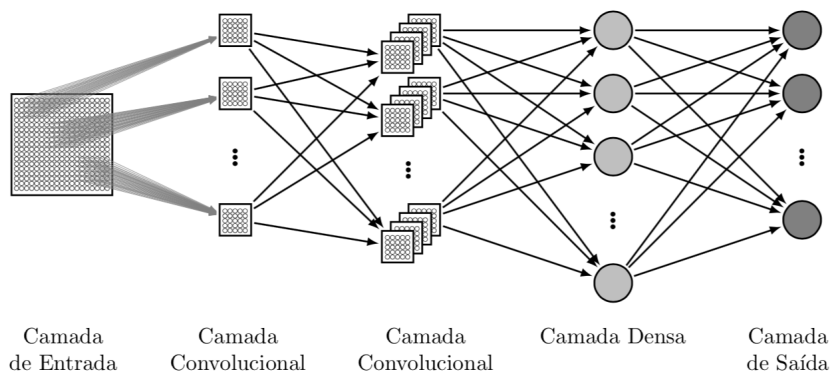
Considerando os resultados promissores já apresentados na literatura da aplicação de redes neurais profundas para problemas de classificação utilizando dados 1-D, este trabalho apresenta os resultados obtidos da pesquisa de aplicação de Redes Neurais Convolucionais Auto-associativas para o problema de classificação de arritmias em sinais ECG. Duas redes neurais convolucionais auto-associativas (CAEs) foram inicialmente treinadas para reproduzir nas suas saídas os sinais ECG apresentados na suas entradas. Cada rede foi treinada para aprender as características dos sinais, sendo uma rede treinada para o caso de pacientes com arritmias e outra para o caso de pacientes com batimentos cardíacos normais. Após treinamento, as duas redes foram arranjadas em uma estrutura paralela competitiva que permite então classificar novos sinais ECG em uma das classes: normal ou com arritmia. Para desenvolvimento e treinamento das redes foi utilizada a base de dados *MIT-BIH Arrhythmia Database* [PhysioNet 2018], que apresenta medidas de sinais ECG de 48 pacientes. Os sinais utilizados para desenvolvimento do sistema foram os sinais ECG brutos sem a necessidade de passar por um processo de extração de características ou de ruídos.

## **2. Redes Neurais Convolucionais Auto-Associativas**

Uma rede neural baseada em Aprendizado Profundo apresenta múltiplas camadas de processamento com a capacidade de aprender representações de dados em múltiplos níveis de abstração [Lecun et al. 2015]. Estas camadas transformam os dados brutos em um nível de representação maior a cada camada, permitindo com que funções mais complexas possam ser aprendidas. Dentre as redes neurais com aprendizado profundo destacam-se as *Long Short Term Memory* (LSTM), as Redes Neurais Auto-Associativas

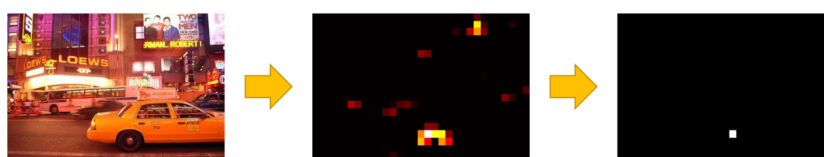
(Autoencoders), as Redes Auto-associativas Empilhadas (*Stacked Autoencoders*) e as Redes Neurais Convolucionais (*Convolutional Neural Network*).

A Rede Neural Convolucional (CNN) é uma arquitetura de Aprendizado Profundo que utiliza filtros convolucionais e camadas de *pooling* para o auxílio na extração de parâmetros dos dados de entrada. No geral, a CNN tenta reproduzir o comportamento de certos neurônios biológicos que respondem a algumas orientações de bordas identificadas pelo olho humano. A Figura 2 exemplifica a arquitetura de uma CNN.



**Figura 2. Representação da arquitetura de uma CNN com duas camadas convolucionais e uma camada densa (camada completamente conectada). Fonte: [Sakurai 2017]**

A CNN possui a camada de convolução que utiliza filtros para obter os parâmetros mais expressivos oriundos das entradas da CNN. Os filtros são compostos por janelas deslizantes que se movimentam através do campo receptivo para a extração de características. A característica a ser identificada é definida através do tipo de filtro (*kernel*) a ser utilizado, como por exemplo, o filtro de identificação de bordas. Ao final do processo, quando a janela deslizante passar por todos os pontos da entrada, tem-se o mapa de ativação. A Figura 3 demonstra esse processo.



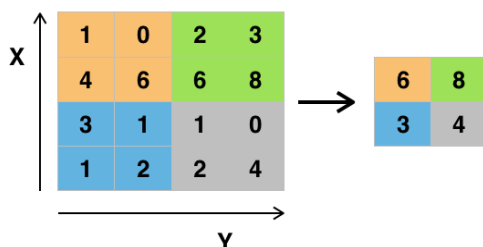
**Figura 3. Processo envolvendo o Mapa de ativação na Camada de Convolução. Imagem original (À esquerda), Mapa de Ativação (Centro). A característica mais latente em destaque (À direita). Fonte: [Deshpande 2017].**

O tamanho do filtro de convolução varia de aplicação para aplicação, por exemplo, em uma tarefa de reconhecimento de imagens o tamanho pode variar em função do número de pixels da janela deslizante, enquanto em um processamento de linguagem natural tem-se a quantidade de palavras para se definir o tamanho da janela deslizante.

Logo após a camada de convolução tem-se uma camada de ativação Retificação Linear (ReLU) como função de ativação, que é comumente usada para a introdução de não-linearidade ao modelo [Nair e Hinton 2010].

Na camada de *Pooling* é realizada a seleção de determinadas característica dos parâmetros já filtrados pela camada de convolução. Esta camada possui duas funções:

reduzir o total de parâmetros e pesos da rede, e controlar o sobre-treinamento da rede [Deshpande 2017]. Uma das opções mais comuns para esta camada é o *maxpooling*, onde se encontra o valor máximo no conjunto de parâmetros resultantes das camadas anteriores. A Figura 4 demonstra o funcionamento desse processo.



**Figura 4. Execução do *maxpooling* em uma matriz de parâmetros divididos em 4 partes iguais. Fonte: [Deshpande 2017].**

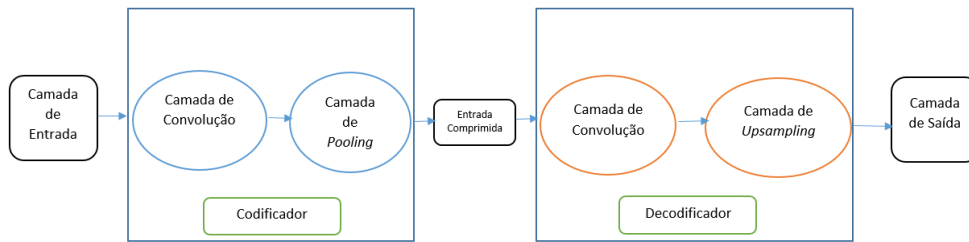
Uma CNN possui após suas camadas de processamento (Convolução e *Pooling*) uma camada completamente conectada (camada densa) que segue o padrão visto em diversas arquiteturas de redes, como por exemplo das Redes Neurais *Perceptron* de múltiplas camadas (MLP) [MLP 2018]. É nesta camada que ocorre a tarefa para a qual a CNN foi designada, podendo ser tarefa de classificação, regressão, etc. Diversos algoritmos podem ser utilizados para o treinamento da CNN, como por exemplo o algoritmo *Backpropagation*, comumente utilizado para treino das MLPs [Sathyanarayana 2014].

Uma rede neural auto-associativa é uma arquitetura de rede focada na reconstrução na sua saída dos sinais apresentados em sua entrada. Em uma rede neural auto-associativa ocorre a compressão dos dados de entrada da rede na camada intermediária, onde então as características dos dados de entrada são aprendidas de forma específica. Uma rede auto-associativa possui duas camadas de processamento: a camada codificadora (*encoder*) para a compressão dos dados e uma decodificadora (*decoder*) para a descompressão deles. A Figura 5 apresenta a estrutura de uma rede auto-associativa.



**Figura 5. Representação de uma rede Auto-associativa.**

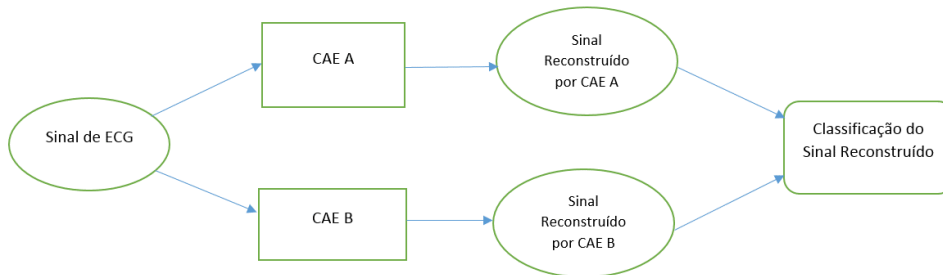
As redes MLPs vêm sendo bastante utilizadas como redes auto-associativas, para aplicação em diversos problemas de classificação [Castro et al. 2012]. Assim como as redes MLPs, as CNNs também podem ser utilizadas para representação típica de uma rede rede neural auto-associativa, tendo-se assim uma CNN auto-associativa (CAE), sendo que a mesma apresentará as camadas de codificação e decodificação em múltiplos níveis de processamento para o processo da extração de características. No caso da camada de codificação esta pode ser formada por diversas camadas de convolução e *pooling*, para que o sinal seja comprimido com suas principais características filtradas. No caso da camada decodificadora as camadas de convolução e *pooling* também podem estar presentes, mas realizando operações de forma contrária, para a reconstrução dos dados de entrada. A Figura 6 apresenta a estrutura de uma rede CNN auto-associativa.



**Figura 6. Representação de uma Rede Convolutiva Auto-associativa.**

### 3. Estrutura Competitiva de Redes Neurais Convolucionais Auto-associativas para Classificação de sinais ECGs

O sistema para classificação de sinais ECG proposto neste trabalho é composto por duas redes CNN auto-associativas (CAE A e CAE B) arranjadas em uma estrutura paralela e trabalhando de forma competitiva. A Figura 7 apresenta a estrutura do sistema proposto.



**Figura 7. Estrutura do sistema de Classificação dos Sinais ECG.**

Para construção da estrutura classificadora, inicialmente, cada CAE foi treinado para reconstrução do sinal ECG apresentado na sua entrada, sendo que o CAE A foi treinado apenas com sinais ECG de pacientes com batimentos cardíacos normais e o CAE B foi treinado com sinais ECG de pacientes com arritmia. Cada CAE, durante o processo de aprendizado, aprendeu as características dos sinais utilizados para treinamento, sendo que após treinamento, quando se apresenta para cada CAE novos sinais/padrões não utilizados no treino, mas que fazem parte da mesma distribuição dos dados usados para treinamento, cada CAE consegue reconstruir o sinal, com um erro médio quadrático entre a entrada/saída com valor pequeno.

Após treinamento, as duas redes CAEs foram arranjadas na estrutura paralela competitiva da Figura 7, sendo que quando um novo sinal ECG a ser classificado for apresentado para a estrutura, as duas redes CAEs tentarão reconstruir, simultaneamente, o sinal nas suas saídas, sendo que uma delas apresentará um menor erro médio quadrático entre entrada/saída (por ter sido treinada com padrões próximos ao padrão que está sendo apresentado na entrada da estrutura), sendo considerado então o CAE vencedor e o sinal classificado de acordo com a classe deste.

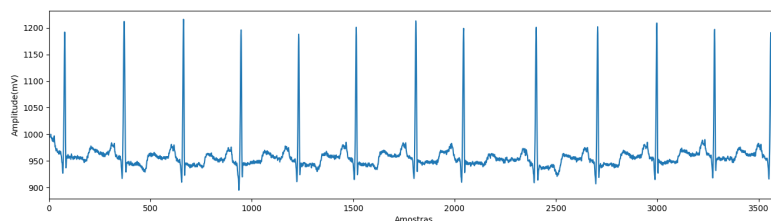
#### 3.1. Base de Dados para treinamento do sistema

Os sinais ECG utilizados para desenvolvimento do sistema foram obtidos do banco de dados *MIT-BIH Arrhythmia Database* [PhysioNet 2018]. Esta base é a base de dados mais representativa de sinais de arritmia disponível para desenvolvimento de métodos de classificação automática e por isso vem sendo largamente utilizada nas pesquisas já publicadas na literatura. A base é composta por sinais ECG obtidos de 48 pacientes,

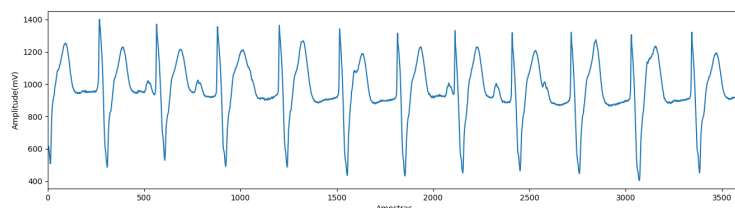
com casos de ritmo cardíaco normal e com arritmia. O sinais foram medidos a uma frequência de 360 Hz, por aproximadamente 30 minutos.

Dos 48 sinais ECG (tem-se dois sinais medidos para um paciente específico), optou-se por excluir dois deles para a implementação do sistema, para que ficassem equilibradas o total de amostras para treinamento das classes, totalizando 23 para cada classe. Os 46 sinais selecionados foram divididos em duas classes: ritmo cardíaco normal e ritmo cardíaco com arritmia.

Para cada sinal de um determinado paciente foram selecionados 10 segundos dentre os 30 minutos medidos, o que gerou 3600 amostras para cada padrão que foi utilizado como entrada para treinamento e teste do sistema. A Figura 8 apresenta um exemplo de sinal ECG com 3600 amostras para um paciente com ritmo cardíaco normal, enquanto a Figura 9 apresenta um exemplo de sinal ECG com 3600 amostras para um paciente com arritmia.



**Figura 8. ECG de um paciente com ritmo cardíaco normal.**



**Figura 9. ECG de um paciente com ritmo cardíaco com arritmia**

### 3.2. Treinamento das Redes Convolucionais Auto-Associativas

Duas CAEs foram treinadas para reconstrução do sinal ECG. Cada rede CAE treinada apresenta 3600 entradas e 3600 saídas relacionadas as 3600 amostras do sinal ECG de cada padrão. A rede CAE A foi treinada apenas com os padrões pertencentes a classe Normal e a rede CAE B foi treinada com padrões pertencentes a classe Arritmia. Ambos os CAEs possuem a mesma estrutura: um *encoder* composto por três camadas de convolução e três camadas de *pooling*, uma intercalando a outra, e um *decoder* com quatro camadas de convolução e três camadas de *upsampling*.

Os sinais de entrada ECG que alimentam cada CAE foram apresentados na forma bruta, sem necessidade de pré-processamento, como filtragem ou detecção de complexo QRS [Isin e Ozdalili 2017]. Todos os sinais de entrada são séries temporais, que alimenta uma rede que contém uma camada de convolução, no qual normalmente realiza uma convolução espacial [Keras 2018], sendo necessário desta forma realizar um pré-processamento nos sinais de entrada para transformá-los de uma dimensão para duas dimensões. Entretanto, o treinamento foi realizado através da linguagem de programação *Python 3.5.3* [Python 2018] utilizando o módulo *Keras*, que permite trabalhar com os sinais em formato de uma dimensão através da função *conv1d* [Keras

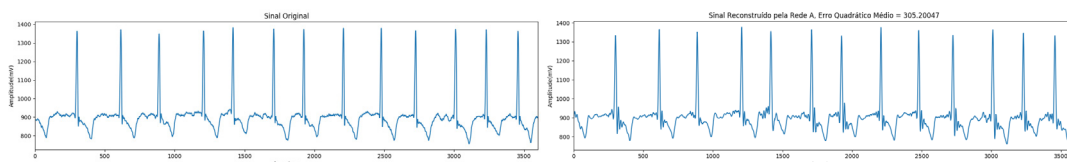
2018]. Cada janela de convolução possui tamanho 20 e aplica *padding* do tipo *same* [Keras 2018]. Essa configuração se repete também para o *decoder* do CAE.

As camadas de *Maxpooling*, a qual está presente no *encoder*, e *Upsampling*, presente no *decoder*, também utilizam de sua versão para dados de séries temporais, *MaxPooling1D* e *UpSampling1D* [Keras 2018] respectivamente, onde o *MaxPooling1D* tem *pool size* de tamanho 2, o qual divide pela metade o tamanho dos dados de entrada para cada camada de *pooling* pela qual eles são processados, comprimindo o total de amostras de cada sinal de 3600 para 450, mantendo o valor máximo encontrado em cada janela de sinal filtrado. O *UpSampling1D* restaura os valores máximos comprimidos, utilizando-se também *pool size* de tamanho 2, e com o auxílio das camadas de convolução, reconstrói o sinal original.

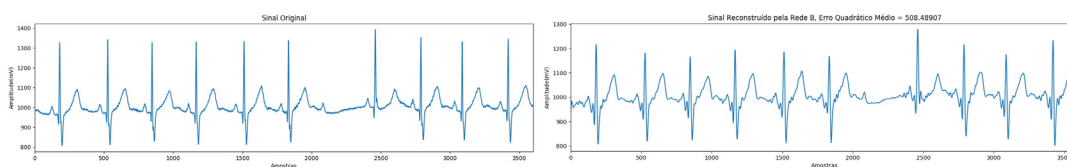
Devido a grande quantidade de amostras nos dados de entrada e a complexidade das operações na camada de *encoder* e *decoder*, as redes foram treinadas com o auxílio de uma GPU GTX 765m com suporte à tecnologia CUDA [NVIDIA 2018], sendo que o módulo *Keras* possui como *backend* outro módulo chamado *TensorFlow* [TensorFlow 2018] que permite esse processo. A plataforma usada para o treinamento também conta com um processador i5-4200m e 8GBs de RAM. Cada rede foi treinada com 7550 épocas, utilizando-se como função de treinamento o *adamax*, que possui uma taxa de treinamento adaptável, a qual se atualiza de forma inversamente proporcional ao seu gradiente [Ruder 2016], sendo o erro para análise de desempenho de cada rede o erro médio quadrático.

A base de dados foi dividida em base de dados para treino e teste do sistema, sendo utilizada 14 amostras para treino e 9 amostras para teste, de cada classe, totalizando 28 amostras de treino e 18 amostras para teste do sistema. Todos os dados foram normalizados na escala entre 0 e 1, e o erro quadrático médio utilizado pra avaliar o desempenho da rede durante o treino calculado dentro desta escala normalizada. Os gráficos gerados pelo sistema tem seu erro quadrático médio calculado para os dados não normalizados.

Para a rede CAE A treinada para a classe Normal, no final de treinamento, para todos os dados de treino, obteve-se um erro médio quadrático de  $2,916283 \times 10^{-4}$ . A Figura 10 apresenta um exemplo de reconstrução do sinal fornecido pela rede para o caso do paciente com ritmo cardíaco normal. Para a rede CAE B treinada para a classe Arritmia, no final de treinamento, para todos os dados de treino, obteve-se um erro médio quadrático de  $2,143963 \times 10^{-4}$ . A Figura 11 apresenta um exemplo de reconstrução do sinal fornecido pela rede para o caso do paciente com arritmia.



**Figura 10. ECG original e reconstruído de um paciente com ritmo cardíaco normal.**



**Figura 11. ECG original e reconstruído de um paciente com arritmia.**

### 3.3. Resultados

Após realizado os treinamentos da CAE A e CAE B. A estrutura da Figura 7 foi organizada e então foi apresentada para a mesma todos os dados da base de teste para classificação. Para verificação do desempenho do sistema classificador utilizou-se as medidas de Acurácia e Sensibilidade, calculadas de acordo com as equações 1 e 2, respectivamente:

$$Acurácia = \frac{Verdadeiros\ Positivos + Verdadeiros\ Negativos}{Verdadeiros\ Positivos + Falsos\ Positivos + Verdadeiros\ Negativos + Falsos\ Negativos} \quad (1)$$

$$Sensibilidade = \frac{Verdadeiros\ Positivos}{Verdadeiros\ Positivos + Falsos\ Negativos} \quad (2)$$

O sistema implementado obteve uma acurácia de 88,9% e sensibilidade de 88,9%, com base nas 18 amostras da base de teste. As 2 amostras classificadas de forma incorreta foram referentes a um paciente pertencente a classe normal e outro pertencente a classe com arritmia. A Figura 12 exibe a matriz de confusão do sistema.

**Matriz de Confusão**

		Normal	Arritmia	
Classe Predita	Normal	8 44.4%	1 5.6%	88.9% 11.1%
	Arritmia	1 5.6%	8 44.4%	88.9% 11.1%
		Normal	Arritmia	88.9% 11.1%
		Classe Desejada		

**Figura 12. Matriz de Confusão gerada para dados de teste.**

A Tabela 1 apresenta alguns resultados de classificadores de sinais ECG já apresentados na literatura usando para desenvolvimento a base de dados MIT. Uma comparação direta de resultados deve ser realizada com cuidado visto que para todos os classificadores apresentados tem-se a base de dados usada para treino e teste diferentes para desenvolvimento dos sistemas. Os resultados são apresentados considerando a base de teste usada por cada autor.



**Tabela 1. Resultados apresentados na literatura.**

Referência	Características da entrada	Classificador	Resultados
[de Chazal et al 2004]	Intervalos de ECG, Morfológico	<i>Weighted LD</i>	Acc = 83% Sen = 87%
[Soria e Martinez 20090]	Intervalo RR, VGG, Morfológico + FFS	<i>Weighted LD</i>	Acc = 90% Sen = 92%
[Llamedo e Martinez 2011]	Wavelet, VCG + SFFS	<i>Weighted LD</i>	Acc = 93% Sen = 95%
[Mar et al. 2011]	Características Temporais, Morfológico, Características Estatísticas + SFFS	<i>Weighted LD e MLP</i>	Acc = 89% Sen = 89%
[Bazi et al. 2013]*	Morfológico, Wavelet	SVM, IWKLR, DTSVM	Acc = 92%
[Ye et al. 2012]	Morfológico, Wavelet, Intervalo RR, ICA, PCA	SVM	Acc=86.4% Sen = 88%
[de Lannoy et al. 2010]	Intervalos de ECG, Morfológico, HOS, Coeficientes HBF	<i>Weighted SVM</i>	Acc = 83% Sen = 80%
[Park et al. 2008]	HOS, HBF	SVM Hierárquico	Acc = 85% Sen = 86%
[Zhang et al. 2014]	Intervalo RR, Morfológico, Segmentos e Intervalos de ECG	SVM Combinado	Acc = 86% Sen = 89%
[Escalona-Moran et al. 2015]	Onda Bruta	RC	Acc = 98% Sen = 96%
[Lin e Yang 2014]**	Intervalo RR Normalizado	weighted LD	Acc = 93% Sen = 91%
[de Lannoy et al. 2012]	Intervalo RR, Segmentos de ECG, Morfológico, HBF, HOS	weighted CRF	Acc = 83%
[Zhang e Luo 2014]	Intervalo RR, Morfológico, Segmentos e Intervalos de ECG, Coeficientes de wavelets	SVM Combinado	Acc = 87% Sen = 88%

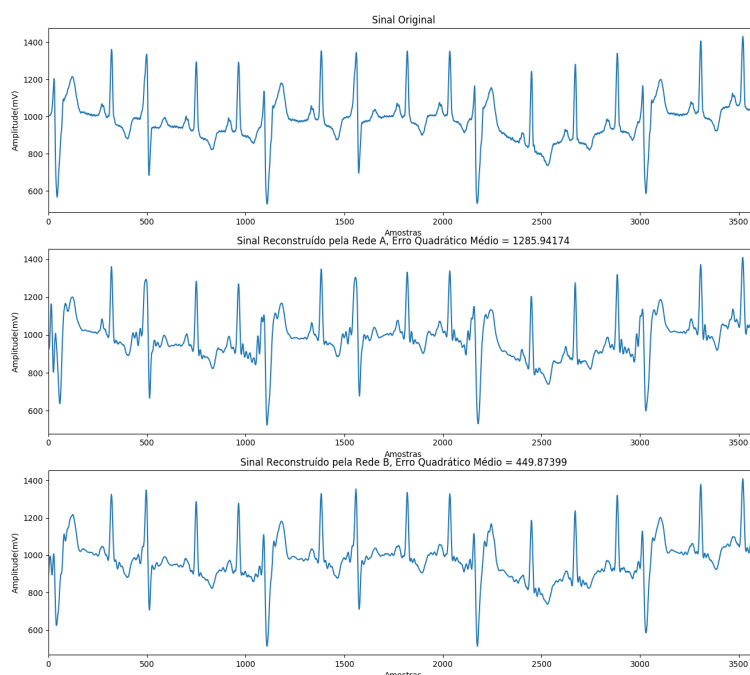
PCA: Primeiro Componente Principal; FFS: *Floating Feature Selection*; ICA: *Independent Component Analysis*; HBF: *Hermite Basis Function*; HOSC: *High Order Statistics Cummulants*; LD: Discriminante Linear; SFFS: *Sequential forward floating search*; IWKLR: *Importance Weighted Kernel Logistic Regression*; CRF: *Conditional Random Fields*; RC: Computação por Reservatório; SVM: *Support Vector Machine*; MLP: Perceptron Multicamada; Acc: Acurácia; Sen: Sensibilidade.

\* Matriz de confusão não disponibilizada, alguns resultados não foram calculados.

\*\* Os Autores otimizam o resultado pra 3 classes (Normal, Ectópica Supraventricular e Ectópica Ventricular).

A Figura 13 exibe um sinal classificado corretamente segundo o sistema proposto, onde o valor “0” representa a classe normal e “1” a classe com arritmia.

Paciente 233m, Classe (Predição) = 1, Classe (Real) = 1



**Figura 13. Exemplo de Sinal classificado pelo Sistema.**

#### 4. Conclusão

Este trabalho apresentou os resultados obtidos da pesquisa de aplicação de Redes Neurais Convolucionais Auto-associativas para o problema de classificação de arritmias em sinais ECG. O sistema pode ser utilizado como sistema de apoio à tomada de decisão por especialistas da área de Cardiologia, sendo uma importante ferramenta para auxílio ao diagnóstico. A implementação apresentada neste artigo obteve uma acurácia de 88,9%, valor próximo ao encontrado em pesquisas recentes documentadas na literatura, e sensibilidade de 88,9%, sendo que esta medida representa a eficácia do diagnóstico em pacientes que possuam de fato alguma anormalidade no ritmo cardíaco.

Cita-se como vantagem do sistema proposto o uso direto do sinal ECG medido, sem a necessidade de uma fase de extração de ruídos ou extração de características, o que diminui o custo computacional do sistema. Para trabalhos futuros, um fator importante a ser aprofundado é seguir o protocolo, já utilizado em muitos trabalhos, relativo à divisão da base de dados MIT para desenvolvimento do sistema classificador, conforme apresentado em [Luz et al. 2015].

#### Referências

- ECG (2018) “ECG Learning Center”, <https://ecg.utah.edu/lesson/1>, Junho.
- Isin, A., Ozdalili, S. (2017) “Cardiac arrhythmia detection using deep learning”, 9th International Conference on Theory and Application of Soft Computing, Computing with Words and Perception, ICSCCW 2017, Budapeste, Hungria.
- Lassoued, H., Ketata, R. (2018) “ECG multi-class classification using neural network as machine learning model”, International Conference on Advanced Systems and Electric Technologies (IC\_ASET), Hammamet, Tunísia, Março.

- Kelwade, J., Salankar, S. (2015) "Prediction of Cardiac Arrhythmia using Artificial Neural Network", International Journal of Applications, Vol. 115, nº 20, Abril.
- Naik, G., Reddy, K. (2016) "Comparative Analysis of ECG Classification Using Neuro-Fuzzy Algorithm and Multimodal Decision Learning Algorithm", 3rd International Conference on Soft Computing & Machine Intelligence, Dubai, Emirados Árabes Unidos, Outubro.
- Tandale, S., Ghongade, R., Barhatte, A., Dale, M. (2017) "Arrhythmia classification using neuro fuzzy approach", 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA), Dehradun, Índia, Setembro.
- Ye, C., Kumar, B., Coimbra, M. (2012) "Combining general multi-class and specific two-class classifiers for improved customized ECG heartbeat classification", International Conference on Pattern Recognition (ICPR), p.2428–2431.
- Escalona-Moran, M., Soriano, M., Fischer, I., Mirasso, C., (2015) "Electrocardiogram classification using reservoir computing with logistic regression", IEEE J. Biomed. Health Inform. 19 (3), p.892–898.
- PhysioNet, PhysioBank (2018) "MIT-BIH Arrhythmia Database". <https://www.physionet.org/physiobank/database/mitdb/>, Junho.
- LeCun, Y., Bengio, Y., Hinton, G. (2015) "Deep Learning Review", Nature: Vol. 521, p. 436-444.
- Nardelli, P., Jimenez-Carretero, D., Bermejo-Pelaez, D., Washko, G., Rahaghi, F., Ledesma-Carbayo, M., Estépar, R. (2018) "Pulmonary Artery-Vein Classification in CT Images Using Deep Learning", DOI 10.1109/TMI.2018.2833385, IEEE Transactions on Medical Imaging.
- Penha, D., Castro, A. (2017) "Convolutional Neural Network Applied to the Identification of Residential Equipment in Nonintrusive Load Monitoring Systems", 3rd International Conference on Artificial Intelligence and Applications, Computer Science & Information Technology, p. 11.
- Sakurai, R. (2017) "Implementando a estrutura de uma Rede Neural Convolutacional utilizando o MapReduce do Spark", <http://rafaelsakurai.github.io/cnn-mapreduce/>, Junho.
- Deshpande. A. (2018) "A Beginner's Guide to Understanding Convolutional Neural Networks", <https://adeshpande3.github.io/adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>, Junho.
- Nair, V., Hinton, G. (2010) "Rectified Linear Units Improve Restricted Boltzmann Machines", Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel.
- MLP, Deep Learning (2018) "Multilayer Perceptron. DeepLearning 0.1 documentation", <http://deeplearning.net/tutorial/mlp.html>, Junho.
- Sathyanarayana, S. (2014) "A Gentle Introduction to Backpropagation, Numeric Insight. 2014.

- Castro, A., Miranda, V., Lima, S. (2012) “Diagnosing Faults in Power Transformers With Autoassociative Neural Networks and Mean Shift”, IEEE Transactions on Power delivery, vol. 27, no. 3, Julho.
- Keras (2018) “Keras: The Python Deep Learning Library”, <https://keras.io/>, Junho.
- Python (2018) “Python”, <https://www.python.org/>, Junho.
- Nvidia (2018) “CUDA Zone”, <https://developer.nvidia.com/cuda-zone>, Junho.
- TensorFlow (2018) “TensorFlow”, <https://www.tensorflow.org/>, Junho.
- Ruder, S. (2016) “An overview of gradient descent optimization algorithms”, <http://ruder.io/optimizing-gradient-descent/index.html#adamax>, Junho.
- de Chazal, P., M., O’Dwyer, M., Reilly, R. (2004) “Automatic classification of heartbeats using ECG morphology and heartbeat interval features”, IEEE Trans. Biomed. Eng. 51 (7), p.1196–1206.
- Soria, M., Martínez, J. (2009) “Analysis of multidomain features for ECG classification”, Comput. Cardiol., p.561–564.
- Llamedo, M., Martínez, J. (2011) “Heartbeat classification using feature selection driven by database generalization criteria”, IEEE Trans. Biomed. Eng. 58 (3), p.616–625.
- Mar, T., Zaunseder, S., Martínez, J., Llamedo, M., Poll, R. (2011) “Optimization of ECG classification by means of feature selection”, IEEE Trans. Biomed. Eng. 58 (8), p.2168–2177.
- Bazi, Y., Alajlan, N., AlHichri, H., Malek, S. (2013) “Domain adaptation methods for ECG classification”, International Conference on Computer Medical Applications (ICCMA), p.1–4.
- de Lannoy, G., François, D., Delbeke, J., Verleysen, M., (2010) “Weighted SVMs and feature relevance assessment in supervised heart beat classification”, Biomedical Engineering Systems and Technologies (BIOSTEC), p.212–223.
- Park, K., Cho, B., Lee, D., Song, S., Lee, J., Chee, Y., Kim, I., Kim, S. (2008) “Hierarchical support vector machine based heartbeat classification using higher order statistics and hermite basis function”, Comput. Cardiol., p. 229–232.
- Zhang, Z., Dong, J., Luo, X., Choi, K., Wu, X. (2014) “Heartbeat classification using disease- specific feature selection”, Comput. Biol. Med. 46, p.79–89.
- Lin, C., Yang, C. (2014) “Heartbeat classification using normalized RR intervals and morphological features”, Math. Problem Eng., p.1–11.
- de Lannoy, G., François, D., Delbeke, J., Verleysen, M. (2012) “Weighted conditional random fields for supervised interpatient heartbeat classification”, IEEE Trans. Biomed. Eng.59 (1), p.241–247.
- Zhang, Z., Luo, X. (2014) “Heartbeat classification using decision level fusion”, Biomed. Eng. Lett. 4 (4), p.388–395.
- Luz, E., Schwartz, W., Cámara-Chávez. G., Menotti, D. (2015) “ECG-based heartbeat classification for arrhythmia detection: A survey”, Elsevier.