Mobile Device ECG Classification using quantized Neural Networks

Sousa Vigno$^{1,2*\dagger}$, Vilson Almeida$^2$, Santos Domingos$^1$, Costa Nator$^3$, Sousa Lopes$^1$ and Pimentel Pedro$^1$

$^*$Correspondence: josevigno@prp.uespi.br

1Universidade Brazil,Rua Carolina Fonseca, 584 - Itaquera, 08230-030, São Paulo - SP, Brasil
Full list of author information is available at the end of the article

$\dagger$Equal contributor

Abstract

In this paper, a novel method for classifying electrocardiogram signals in mobile devices is proposed, which classifies different arrhythmias according to the Association for the Advancement of Medical Instrumentation standard EC57. A convolutional neural network has been constructed, trained and validated with the MIT-BIH Arrhythmia Dataset, which has 5 different classes: normal beat, supraventricular premature beat, premature ventricular contraction, fusion of ventricular and normal beat, unclassifiable beat. Once trained and validated, the model is subjected to a post-training quantization stage using the TensorFlow Lite conversion method. The obtained results were satisfactory, before and after the quantization, the convolutional neural network obtained an accuracy of 98.5%. With the quantization technique it was possible to obtain a significant reduction in model size, thus enabling the development of the mobile application, this reduction was approximately 90% compared to the original model size.

Keywords: ECG; Quantization; CNN; Mobile

1 Introduction

Cardiovascular diseases are one of the leading causes of death worldwide. This type of pathology affects the cardiovascular system, specifically the blood vessels and the heart. There is a growing rate of the population suffering from cardiovascular diseases, including arrhythmia [1]. Sudden death caused by cardiac arrhythmia is a major public health problem worldwide, accounting for 15% to 20% of all deaths. It is estimated that 180,000 to 300,000 sudden cardiac deaths occur in the US annually [2].

In the clinical diagnosis of heart disease, arrhythmia indicates a severe change in heartbeat function and may cause a stroke or sudden cardiac death if left untreated [3]. For detection of cardiac anomalies, the use of electrocardiogram (ECG) is a fundamental diagnosis. This clinical analysis can perform the monitoring of cardiovascular diseases, where this procedure is done manually. However, there are several problems with manually analyzing ECG signals, such as: similarity to other time series data, difficulty in detecting and categorizing different waveform and signal morphology. For a human being, this task is time-consuming and error-prone [4], [5].
To solve problems with manual ECG signal analysis, many studies in the literature use machine learning techniques to accurately detect signal anomalies [6], [7], [8]. However, in these approaches a final application to the learning models is not proposed, a final application is important because the death rate caused by these anomalies is high, as already mentioned.

Technological advances and increased computational power make possible to use techniques that can assist in clinical diagnosis, such as artificial bioinspired Neural Networks in human neurons. As an example, Convolutucional Neural Networks (CNNs) [9], [10]. But CNNs require a high computational cost in end applications, which makes the development of these applications a challenging task.

This paper presents a mobile ECG signal classification system capable of classifying different arrhythmia according to the Advancement of Medical Instrumentation (AAMI) EC57 standard [11], [12]: normal beat, supra-ventricular premature beat, premature ventricular contraction, fusion of ventricular and normal beat, unclassifiable beat.

The proposed method consists in the implementation of a CNN for the arrhythmia classification task, the model was trained and tested using the MIT-BIH Arrhythmia Dataset [13], containing 109,449 ECG signal samples. After training and validation, the model was submitted to a quantization stage, where the post-training quantization technique was used, using the TensorFlow Lite (TFL) [14] conversion method. This technique is to drastically reduces model size, power consumption and processing, enabling the development of a mobile application using a CNN.

This article is divided into seven sections. In section 2, it consists of a summary of the literature. Section 3 describes the data set as well as its analysis, while section 4 presents the methodology and its steps. In section 5 describe the evaluation metrics of the neural network, 6 in turn, is about the results and discussions, and finally present the conclusion in section 7.

2 Related Work

Analyzing the state of the art, it is possible to notice the increasing number of cardiac signal analyzes, as an example of Wang et al. [15], which aims to propose a method of heart rate ECG identification, suitable for short-term signals. The method aims to completely preserve the original temporal and morphological information of the QRS complex, solving the problem of T-wave displacement. They use the Principal Component Analysis Network.

Berkaya et al. published an ECG survey, which consists of a literature review related to ECG analysis [16], considering the following aspects: preprocessing, resource extraction, resource selection, resource transformation, classification, application fields, databases and success measures, is also mentioning the following: most used classifiers in the literature.

Lu et al. proposed a classification system with generalization capacity [17], where a resource extraction and balancing through the Random Sampler Algorithm was used, as a classifier was used the Random Forest, thus obtaining precision results above 99.0%.

ECG signal processing techniques for real time analysis are implemented in Raj at al. [18] and Varatharajan et al. [19], They use the Support Vector Machine algorithm
for pattern recognition. These methods can be used for screening and pathological classifications, as well as a weighted kernel to identify Q, R, and S waves at the ECG signal input to classify the pulse level.

In [20], Zihlmann et al. explains the use of two Deep Neural Networks (DNNs) architectures for ECG classification by assessing the atrial fibrillation classification data set provided by PhysioNet/CinC Challenge 2017. The first architecture is a CNN with mean-to-length feature aggregation over time and second is a convolutional recurrent neural network that combines a 24-layer CNN with a long-short-term-memory 3-layer network for temporal aggregation of features.

In [21], Hannui et al. demonstrated the implementation of a DNN to classify 12 heart rate classes using 91,232 single-lead ECGs from 53,549 patients who used a single electrode ambulatory ECG monitoring device. DNN reached an average area under the characteristic operating curve of 0.97.

In [3], Yang et al. proposes a new method of arrhythmia classification through ECG with Statement on Standards for Attestation Engagements and a softmax regression model. The algorithm is employed to hierarchically extract high level resources from the huge amount of ECG data.

The article of Xia et al. [22], features an automatic wearable ECG classification and monitoring with a stack denoization autoencoder system. Using a wireless sensor device to retrieve ECG data and send that data to a Bluetooth 4.2 computer, where softmax regression is used to rate the ECG beats.

The work of Shaker et al. [23], proposes a new data augmentation technique using Generative Adversary Networks (GANs), to balance the classes of the MIT-BIH Arrhythmia Dataset. They use two deep learning approaches: a) An end-to-end approach and a two-stage hierarchical approach, b) Based on CNNs. The results show that, with the increase in the data generated through the proposed technique, it is possible to observe an effective improvement in the performance of ECG classification in relation to the classification of the original dataset.

The article of Gao et al. [24], implements a Long Short Term Memory (LSTM) neural network to use the timing features in ECG signals, Focal Loss (FL) is used to resolve the imbalance of the MIT-BIH arrhythmia database. The results show that the LSTM network with FL obtains an accuracy of 99.26%.

In Kumar et al. [25], it presents a method for ECG classification, which employs a generalized signal pre-processing technique and uses a Multi-layered Perceptron network for the arrhythmia classification task, according to the AAMI EC57 standard. The method is trained and evaluated using PhysioNet MIT-BIH data set, obtaining an average accuracy of 98.72%.

The article of Liu et al. [26], proposed an algorithm based on CNN capable of classifying cardiac arrhythmia using the ECG. Comparing different CNN models and then using them to increase the correctness rate in the classification, combining Linear Discriminant Analysis (LDA) and the support Vector Machine (SVM). The MIT-BIH Arrhythmia Database was also used in this work, where the authors achieved a result of 91.29% correct.

3 Dataset Description and Analysis

This collection used a collection of heartbeat signals from the open-source database, MIT-BIH Arrhythmia Dataset [13], with 109,449 samples from 47 individuals, with
a sampling frequency of 360Hz and about 48 hours of recording, the longest beats have 260 values. The samples are cropped, reduced or filled with zeros if necessary for a fixed dimension of 187 values.

There was also preprocessing in the MIT-BIH dataset, creating a new heart rate table divided into 5 new categories according to the AAMI EC57 standard [11], [12], whose are: Normal, Supra-Ventricular Premature Beat, Ventricular Escape, Fusion of Ventricular and Normal, Unclassifiable Beat.

The data collected consists of samples of ECG signals that had a recording time of 1400 milliseconds. The number of samples is divided into 5 classes, namely: class 0 with 90,589, class 1 with 2,779, class 2 with 7,236, class 3 with 803 and class 4 with 8,039 samples that can be observed in figure 1.

![Figure 1 Data number for each class.](image)

Given the numerical representation of the quantity of each record of the above classes, we have:

- **Class 0**:
  - Normal beat:
    That would be a sign of ECG from a person with their heart health under normal and healthy conditions.

- **Class 1**:
  - Supraventricular premature beat:
    It represent premature activation of the atria from a different site from the sinus node and may originate from the atria or atrioventricular node (called premature junctional beats) [27].

- **Class 2**:
  - Premature ventricular contraction:
    It is an event where the heartbeat is initiated by the Purkinje fibers in the ventricles rather than the sinoatrial node, the normal heartbeat initiator [28].
• **Class 3:**
  - Fusion of ventricular and normal beat:
    A fusion beat occurs when electrical impulses from different sources act on the same region of the heart at the same time [29].

• **Class 4:**
  - Unclassifiable beat:
    Beats that have not been associated with any rating.

Each class presented consists of signals of different types of spectra, i.e., different graphical representation of the intensity distribution of a wavelength from a given signal. In figure 2, each class of ECG signal is represented, containing ten samples for each label.

![Figure 2: Ten samples of signals for each label.](image)

In signal analysis, there are some ways that can alter and change how the signal is viewed, aiding in observations and interpretations. One of the ways that is widely used is the signal spectrum, which is the representation of the components in a graph that shows their amplitude versus frequency.

To obtain a spectral signal, the Discrete Fourier Transform (DFT) is used, which is one of the main tools for analysis of digital signals. DFTs are extremely useful as they reveal the frequencies of the input data as well as the relative forces of any periodic components of a given signal. From this, given a certain discrete time...
signal $x[n]$, its frequency response can be determined using the DFT, which can be given according to the equation 1.

$$X(\omega_k) \triangleq \sum_{N=0}^{N-1} x(t_n)e^{-j\omega_k t_n}$$ \hspace{1cm} (1)

Where $\omega_k$ is the frequency of the sample and $t_n$ is the instant of time in seconds, being $n \geq 0$. $X(\omega_k)$ is the spectrum of $x$ at frequency $\omega_k$ and $x(t_n)$ is defined as the input of the signal amplitude over time $t_n$.

4 Methodology

This section presents the methods used for mobile application development, and describes the metrics used to evaluate the performance of the implemented neural network. The diagram illustrated in Figure 3 shows the main constituent parts of the method. It consists of four main modules: a) CNN training; b) Model quantization using TensorFlow Lite [30], which aims to optimize the model for mobile application and d) Development of Android application for final classification of ECG signals.

4.1 Neural network training and architecture

According to [31], to train a machine learning model, it is necessary to divide the data into two sets (training and testing). The training dataset is the data sample used to fit the model, where the model sees and learns from this data [32]. The test data set, however, is the data sample used to provide an unbiased evaluation of the model in the training data set after adjusting the model hyperparameters [32].

To perform the neural network training, the data set was divided into training and testing, with 70% of each data class used for training and 30% used for testing. Furthermore, the amount of samples for training was 76,614 while for testing is 32,834 samples.

As mentioned in section 3, not all signs are the same length, to treat this problem, samples are cut, reduced or filled with zeros, if necessary, for a fixed dimension of
187 values. After this pre-processing, all signals go through a transformation section with the DFT. Thus, input the neural network are vectors of a dimension with a length of 187, after they have passed through a DFT.

Figure 4 illustrates the proposed network architecture for the arrhythmia classification task. This architecture is based on the architecture proposed by [12]. All convolution layers apply 1D convolution and each has 32 kernels of size 5. Max pooling with size 5 and strides 2 are also used on all pooling layers. The predictor network consists of five residual blocks followed by three fully connected layers with 512, 256, 5 neurons respectively and one softmax layer to predict the output class. The loss function is used with Cross Entropy, to map the network output the Softmax probabilistic function is used.

In order to train the model, the Adam Optimizer is used [33], with a learning rate of 0.0001, the Batch Size of 200 samples, the number of Epochs used is 100 and the Training Time of the network was 13 minutes.

For the implementation of the neural network, the computer library TensorFlow is used [30]. Processing was performed using a GeForce GTX 1060 graphics card with 1280 CUDA cores (processors), 6 GB of dedicated memory, 12 GB of RAM and a fourth generation Core i5 processor.

4.2 Quantized Neural Networks

Quantized Neural Networks use low precision weights and activations. These networks are trained from zero to arbitrary fixed point accuracy. Where in precision, QNNs that use fewer bits require deeper and wider network architectures than networks that use more precise operators, while requiring less complex arithmetic and fewer bits by weight [34].

A method has been introduced to train extremely low precision QNNs with weights and activations (eg 1 bit) at run time. In train-time, quantized weights and activations are used to calculate parameter gradients. During the next steps, QNNs dramatically reduce memory size and access by replacing most arithmetic operations with bit-by-bit operations [35]. This is to say that the quantization scheme is an integer mapping $q$ to real numbers $r$, that is, as follows:

$$r = S(q - Z)$$  \hspace{1cm} (2)
This scheme consists in the multiplication of two square arrays $N \times N$ of real numbers, $r_1$, $r_2$, with its product represented by $r_3 = r_1 r_2$. We denote the entries of each of these matrices $r_\alpha (\alpha = 1, 2, or 3)$ as $r_\alpha^{(i,j)}$ for $i < i,j < N$, and the quantization parameters with which they are quantified as $(S_\alpha, Z_\alpha)$. We denote the inputs quantized by $q_\alpha^{(i,j)}$. Then, equation 2 becomes:

$$r_\alpha^{(i,j)} = S_\alpha (q_\alpha^{(i,j)} - Z_\alpha)$$

(3)

From the definition of matrix multiplication we have:

$$S_3 (q_3^{(i,k)} - Z_3) = \sum_{j=1}^{N} S_1 (q_1^{(i,j)} - Z_1) S_2 (q_2^{(j,k)} - Z_2))$$

(4)

which can be rewritten as:

$$q_3^{(i,k)} = Z_3 + M \sum_{j=1}^{N} (q_1^{(i,j)} - Z_1)(q_2^{(j,k)} - Z_2))$$

(5)

For this application post-training quantization [30] was performed, thereby reducing the model size while improving CPU latency, with little degradation in [30] model accuracy. These techniques can be performed on a trained TensorFlow model and applied during TFL conversion.

There are several post-training quantization options [30], as can be seen from Table 1, the methods chosen and implemented are Weights Hybrid quantization and Full Integer quantization of weights and activations. Weight quantization is the simplest method of Post Training quantization, only floating point weights are quantized to 8 bits precision (also called ”hybrid” quantization) [30]. This technique is performed with the model to TFL converter. The full integer quantization of weights and activations results in a fully quantized model, but the model still uses floating point input and output [30].

Table 1 Post training quantization options.

| Technique            | Benefits                        | Hardware         |
|----------------------|---------------------------------|------------------|
| Post training hybrid | 4x smaller, 2-3x speedup, accuracy | CPU              |
| Post training integer| 4x smaller, More speedup        | CPU, Edge TPU, etc. |
| Post training fp16   | 2x smaller, Potential GPU acceleration | CPU/GPU          |

4.3 Mobile application development

The method chosen by this paper uses the TFL Java API [30], in figure 5, where you can develop applications for Android and IOS. TFL is TensorFlow’s solution with lightweight mobile and embedded models.

The application was developed on the Android platform, aiming at the classification of ECG signals. Data entry consists of loading a text file containing the signal to be analyzed and, after the signal is loaded, it will pass through the neural
network where it will be classified. The output will be given, making available its class in which it was classified by the network with its accuracy. The application interface can be observed according to the figures: 6, 7.

5 Metrics of the evaluation

The final accuracy of the model is estimated by the equation, where \( Ac_f \) is the sum of the differences between the actual value \( y_i \) and the expected value \( \hat{y}_i \) with this it is possible to infer the generalization of the network.

\[
Ac_f = \sum_{i=1}^{k} (y_i - \hat{y}_i)
\]  

As a statistical tool, we have the confusion matrix that provides the basis for describing classification accuracy and characterizing errors, helping to refine accuracy. The confusion matrix is formed by an array of squares of numbers arranged in rows and columns that express the number of sample units of a given category, inferred by a decision rule, compared to the current category [36].

The measurements derived from the confusion matrix are: total precision, which was chosen by the present work, individual class precision, producer precision, user precision, Kappa index, among others.

The total precision is calculated by dividing the sum of the main diagonal of the error matrix, that is, the total sum of the correct predictions, by the total number of samples collected.

As a statistical tool to evaluate the performance of the model, precision and recall are also used. Where precision is given by dividing the numbers of true positives, by the sum of true positives and false positives. The recall is calculated by dividing the true positives, by the sum of the true positives and false negatives.

F1 Score is a simple metric that takes Precision and Recall into account. This is simply the harmonic medium of precision and recall [37].

\[
F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}
\]  

AUC - ROC Curve is a performance measure for classifying problems in various threshold configurations. ROC is a probability curve and AUC represents the degree or measure of separability [38].

6 Results and Discussion

This section discusses and presents the results obtained at each stage of the development of this article. In Table 2, a comparison is made with the works Kachuee et al. [12], Acharya et al. [39]. It is noteworthy that the performance of each quantization method used is demonstrated, together with the result of the developed mobile application. Figure 8, is the confusion matrix before the quantization section and the training progress can be analyzed in figure 9.

| Work              | Approach | Accuracy | ROC AUC | Kappa  | Recall | Precision | F1 Score |
|-------------------|----------|----------|---------|--------|--------|-----------|----------|
| This paper        | CNN      | 98.5%    | 94.4%   | 95.4%  | 92.1%  | 96.6%     | 94.4%    |
| Kachuee et al. [12]| CNN      | 93.4%    | -       | -      | -      | -         | -        |
| Acharya et al. [39]| CNN      | 93.5%    | -       | -      | -      | -         | -        |

ECG classification results are compared with three other studies using the same MIT-BIH dataset, Table 2. In [12], it is proposed a CNN method for heart rate classification, this study had an average accuracy of 93.4% in the arrhythmia classi-
fication. In addition, a method for transferring the acquired knowledge to the task of classifying myocardial infarction is suggested.

In reference [39], a 9-layer CNN is developed to automatically identify 5 different categories of ECG heartbeat with an accuracy of 93.5%. In Table 2, it is possible to see that the performance of the method proposed by the present work surpasses the accuracy of classification of Kachuee et al. [12] and Acharya et al. [39]. It is also noteworthy that both comparative works do not implement a final application for the proposed CNNs. Already this article develops a mobile application to assist the diagnosis of different arrhythmias.

Table 3  Results of metrics used to evaluate neural network performance after Hybrid quantization for each class.

| Classes                              | ROC AUC | Cohen’s Kappa | Recall | Precision | F1 Score |
|--------------------------------------|---------|---------------|--------|-----------|----------|
| Normal beat                          | 98.3%   | 99.0%         | 99.9%  | 99.6%     | 99.9%    |
| Supraventricular premature beat      | 85.5%   | 89.3%         | 88.1%  | 94.7%     | 91.3%    |
| Premature ventricular contraction    | 98.4%   | 97.6%         | 97.6%  | 96.1%     | 97.7%    |
| Fusion of ventricular and normal beat| 83.4%   | 86.1%         | 80.8%  | 87.9%     | 85.1%    |
| Unclassifiable beat                  | 98.7%   | 99.7%         | 99.0%  | 99.9%     | 99.0%    |

On the figure 8, training progress is shown, as it is possible to analyze network performance over each season. It is also noted that test accuracy is similar to training accuracy.

As mentioned in section 4.2, two methods of quantization were implemented: Hybrid quantization of Weights and the Integer quantization of Weights and acti-
The size of the model before the quantization section was 10.2 MB, after the quantization the model started to have the size 3.4 MB with the Hybrid quantization, for the Integer total quantization the model stayed with the size of 862.0 KB. With this it is possible to notice the reduction in the model size after quantization. This significant reduction in the size of the model is crucial for the development of the proposed mobile application, as it also allows a reduction in the computational cost required for the application to work on a mobile device.

On the Table 4, the accuracy of the neural network after quantization is displayed, so it is possible to analyze the accuracy of each quantization method. To evaluate the efficiency of the model, it is tested with different samples quantities, it can be noticed that the variation is small in both the quantization methods. Moreover, these preliminary results made it possible to think of an efficient Android application, with a simple and intuitive user interface, capable of performing the classification
### Table 4
Accuracy after quantization with different sample quantities, using the two quantization methods (Hybrid and Integer).

| Quantity of samples | Accuracy Hybrid | Accuracy Integer |
|---------------------|-----------------|------------------|
| 6000                | 99.11%          | 99.06%           |
| 12000               | 99.03%          | 99.00%           |
| 18000               | 99.02%          | 98.99%           |
| 24000               | 99.05%          | 99.04%           |
| 30000               | 99.06%          | 99.05%           |

### Table 5
Results of metrics used to evaluate neural network performance after Integer quantization for each class.

| Classes                             | ROC AUC | Cohen’s Kappa | Recall | Precision | F1 Score |
|-------------------------------------|---------|---------------|--------|-----------|----------|
| Normal beat                         | 97.5%   | 98.9%         | 99.0%  | 99.4%     | 99.9%    |
| Supraventricular premature beat     | 83.4%   | 88.3%         | 87.0%  | 91.1%     | 90.5%    |
| Premature ventricular contraction   | 98.2%   | 96.2%         | 97.4%  | 97.3%     | 97.3%    |
| Fusion of ventricular and normal beat | 83.5%   | 85.4%         | 80.5%  | 88.0%     | 84.2%    |
| Unclassifiable beat                 | 98.4%   | 99.2%         | 99.5%  | 99.9%     | 99.4%    |

of 5 different arrhythmias. Our goal is to facilitate the use, mobility and low cost classification of different arrhythmias.

As can be seen from figure 1, the dataset is unbalanced with more than 60% of signal samples belonging to class 0 (Normal beat). To analyze the model efficiency against the classification of each class, the model evaluation metrics for each class are used. In Tables 3, 5, it is possible to analyze the results of the metrics for each class both Hybrid and Integer quantization methods, as it can be observed that the model performed well even with the unbalanced dataset.

According to the Figures 6, 7, can be observed that the interface and home screen of the application. In the beginning, it has a graphic where will be visualized the signal, which was loaded by the user. The x-axis of the graph represents the time at which the analyzed signal was recorded. Y axis, in turn, consists of the amplitude of the signal. The application has a simple and intuitive interface which works as follows: to load a signal, the user must click on the LOAD button, which will be directed to the Android file explorer, to select the desired signal and that it should be in csv text format. With the signal loaded, the application is ready for classification, which will be given after clicking the CLASSIFY button. The result will be displayed below the signal graph, which will have the class to which the signal was classified and below will have the prediction percentage.

In the execution of the application, some steps occur in which it is of great importance for the progress of the same. Thus, for comparison purposes, the two methods of quantization of the neural network were used. Where the first test is Hybrid Quantization, which had a runtime for application loading of 38.49 ms, using 26.7 MB RAM and 12% CPU, for classification, got a time response time 36.46 ms. In the second method consisting of Total Quantization, a time of 7.97 ms for application loading was achieved using 17.6 MB ram and 9.5% CPU. The smartphone used for testing was the Zenfone 4, which has a 4x 2.2 GHz + 4x 1.8 GHz processor and 4 GB of RAM.

### 7 Conclusions
In this article, we proposed a novel mobile application capable of classifying 5 different types of heartbeat, where the first class is normal and the others arrhythmias. This method yielded quite satisfactory results, with more than 98% accuracy. A
comparison was also made with two other works in the literature, where the present article obtained superior results in relation to them.

It is also noteworthy that both comparative works do not implement a final application for the proposed CNNs. Already this article develops a mobile application to assist the diagnosis of different arrhythmias. The neural network, in turn, was submitted to Quantization steps, where the Total Quantization method showed greater efficiency, reducing the size of the model by 90%, with a time of 7.97ms for application loading action, using 17, 6MB of RAM and 9.5% CPU.

Abbreviations
AAMI: Advancement of Medical Instrumentation; DFT: Discrete Fourier Transform; CNN: Convolutional neural network; TFL: TensorFlow Lite; ECG: Electrocardiogram.

Availability of data and materials
The dataset used during the current study is freely available at:
https://www.physionet.org/content/mitdb/1.0.0/
https://www.physionet.org/content/ptbdb/1.0.0/

Competing interests
The authors declare that they have no competing interests.

Funding
Not applicable.

Author’s contributions
SJ, VR, and SD designed the methods, performed the experiments, and wrote this article. CN designed the methods and offered help with the project. SL and PP were responsible for the treatment, data analysis and writing of this article.

Acknowledgements
Not applicable.

Author details
1 Universidade Estadual do Piauí, UESPI, Avenida Marechal Castelo Branco, 180, 64260000,Piripiri-PI, Brazil.
2 Universidade Brazil, Rua Carolina Fonseca, 584 - Itaquera, 08230-030, São Paulo - SP, Brasil. 3 Universidade Estadual do Ceará, Av. Dr. Silas Munguba, 1700 - Itaperi, 60714-903 Fortaleza - CE, Brasil.

References
1. Xiong, Z., Stiles, M.K., Zhao, J.: Robust ecg signal classification for detection of atrial fibrillation using a novel neural network. In: 2017 Computing in Cardiology (CinC), pp. 1–4 (2017). IEEE
2. Srinivasan, N.T., Schilling, R.J.: Sudden cardiac death and arrhythmias. Arrhythmia & electrophysiology review 7(2), 111 (2018)
3. Yang, J., Bai, Y., Lin, F., Liu, M., Hou, Z., Liu, X.: A novel electrocardiogram arrhythmia classification method based on stacked sparse auto-encoders and softmax regression. International Journal of Machine Learning and Cybernetics 9(10), 1733–1740 (2018)
4. Dey, N., Ashour, A.S., Shi, F., Fong, S.J., Sherratt, R.S.: Developing residential wireless sensor networks for ecg healthcare monitoring. IEEE Transactions on Consumer Electronics 63(4), 442–449 (2017)
5. Saraiva, A., Miranda de Jesus Castro, F., Ferreira, N., Valente, A.: Compression of electrocardiographs comparative study between the walsh hadamard transform and discrete cosine transform. (2018)
6. Marts, R.J., Acharya, U.R., Lim, C.M., Mandana, K., Ray, A.K., Chakraborty, C.: Application of higher order cumulant features for cardiac health diagnosis using ecg signals. International journal of neural systems 23(04), 1350014 (2013)
7. Li, T., Zhou, M.: Ecg classification using wavelet packet entropy and random forests. Entropy 18(8), 285 (2016)
8. Acharya, U.R., Fujita, H., Oh, S.L., Hagiwara, Y., Tan, J.H., Adam, M.: Application of deep convolutional neural network for automated detection of myocardial infarction using ecg signals. Information Sciences 415, 190–198 (2017)
9. Saraiva, A., Ferreira, N., Sousa, L., Carvalho da Costa, N., Sousa, J., Santos, D., Soares, S.: Classification of images of childhood pneumonia using convolutional neural networks. In: 6th International Conference on Bioimaging, pp. 112–119 (2019)
10. Saraiva, A.A., Santos, D.B.S., Costa, N.J.C., Sousa, J.V.M., Ferreira, N.M.F., Valente, A., Soares, S.F.S.P.: Models of learning to classify x-ray images for the detection of pneumonia using neural networks. In: BIOIMAGING (2019)
11. for the Advancement of Medical Instrumentation, A., et al.: Testing and reporting performance results of cardiac rhythm and st segment measurement algorithms. ANSI/AAMI EC38 1998 (1998)
12. Kachuee, M., Fazeli, S., Sarrafzadeh, M.: Ecg heartbeat classification: A deep transferable representation. In: 2018 IEEE International Conference on Healthcare Informatics (ICHI), pp. 443–444 (2018). IEEE
13. Goldberger, A.L., Amaral, L.A., Glass, L., Hausdorff, J.M., Ivanov, P.C., Mark, R.G., Mietus, J.E., Moody, G.B., Peng, C.-K., Stanley, H.E.: Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. Circulation 101(23), 215–220 (2000)

14. Jacob, B., Klugs, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., Kalenichenko, D.: Quantization and training of neural networks for efficient integer-arithmetic-only inference. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2704–2713 (2018)

15. Wang, D., Si, Y., Yang, W., Zhang, G., Liu, T.: A novel heart rate robust method for short-term electrocardiogram biometric identification. Applied Sciences 9(1), 201 (2019)

16. Berkaya, S.K., Uysal, A.K., Gunal, E.S., Ergin, S., Gunal, S., Gulmezoglu, M.B.: A survey on ecg analysis. Biomedical Signal Processing and Control 43, 216–235 (2018)

17. Lu, W., Hou, H., Chu, J.: Feature fusion for imbalanced ecg data analysis. Biomedical Signal Processing and Control 41, 152–160 (2018)

18. Raj, S., Ray, K.C.: Ecg signal analysis using dct-based dset and pso optimized svm. IEEE Transactions on instrumentation and measurement 66(3), 470–478 (2017)

19. Varatharajan, R., Manogaran, G., Priyan, M.: A big data classification approach using lda with an enhanced svm method for ecg signals in cloud computing. Multimedia Tools and Applications 77(8), 10195–10215 (2018)

20. Zihlmann, M., Perekrestenko, D., Tschannen, M.: Convolutional recurrent neural networks for electrocardiogram classification. In: 2017 Computing in Cardiology (CinC), pp. 1–4 (2017). IEEE

21. Hariyadi, A.Y., Rajpurkar, P., Haghpanahi, M., Tison, G.H., Bourn, C., Turakhia, M.P., Ng, A.Y.: Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature medicine 25(1), 65 (2019)

22. Xia, Y., Zhang, H., Xu, L., Gao, Z., Zhang, H., Liu, H., Li, S.: An automatic cardiac arrhythmia classification system with wearable electrocardiogram. IEEE Access 6, 16529–16538 (2018)

23. Shaker, A.M., Tantawi, M., Sheedeed, H.A., Tolba, M.F.: Generalization of convolutional neural networks for ecg classification using generative adversarial networks. IEEE Access 8, 35592–35605 (2020)

24. Gao, J., Zhang, H., Lu, P., Wang, Z.: An effective lstm recurrent network to detect arrhythmia on imbalanced ecg dataset. Journal of healthcare engineering 2019 (2019)

25. Kumar, G., Pawar, U., O’Reilly, R.: Arrhythmia detection in ecg signals using a multilayer perceptron network (2019)

26. Liu, J., Song, S., Sun, G., Fu, Y.: Classification of ecg arrhythmia using cnn, svm and lda. In: International Conference on Artificial Intelligence and Security, pp. 191–201 (2019). Springer

27. Kochhäuser, S., Dechering, D.G., Dittrich, R., Reinke, F., Ritter, M.A., Ramtin, S., Duning, T., Frommeyer, G., Eckardt, L.: Supraventricular premature beats and short atrial runs predict atrial fibrillation in continuously monitored patients with cryptogenic stroke. Stroke 45(3), 884–886 (2014).

doi:10.1161/STROKEAHA.113.003788.

https://www.ahajournals.org/doi/pdf/10.1161/STROKEAHA.113.003788

28. Salavatian, S., Yamaguchi, N., Hoang, J., Lin, N., Patel, S., Ardell, J.L., Armour, J.A., Vaseghi, M.: Premature ventricular contractions activate vagal afferents and alter autonomic tone: implications for premature ventricular contraction-induced cardiomyopathy. American Journal of Physiology-Heart and Circulatory Physiology 317(3), 607–616 (2019). doi:10.1152/ajpheart.00286.2019. PMID: 31322427.

https://doi.org/10.1152/ajpheart.00286.2019

29. Conover, M.B.: Understanding Electrocardiography. Elsevier Health Sciences, ??? (2002)

30. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: Tensorflow: A system for large-scale machine learning. In: 12th (USENIX) Symposium on Operating Systems Design and Implementation (OSDI) 16, pp. 265–283 (2016)

31. Shahin, M.A., Maier, H.R., Jaksa, M.B.: Data division for developing neural networks applied to geotechnical engineering. Journal of Computing in Civil Engineering 18(2), 105–114 (2004)

32. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence 5(4), 221–232 (2016)

33. Kingma, D.P., Da, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

34. Moons, B., Goetschalckx, K., Van Berckelaer, N., Verhelst, M.: Minimum energy quantized neural networks. In: 2017 51st Asilomar Conference on Signals, Systems, and Computers, pp. 1921–1925 (2017). IEEE

35. Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., Bengio, Y.: Quantized neural networks: Training neural networks with low precision weights and activations. The Journal of Machine Learning Research 18(1), 6869–6898 (2017)

36. Saraf, A., Melo, R., Filipe, V., Sousa, J., Ferreira, N.F., Valente, A.: Mobile multirobot manipulation by image recognition (2018)

37. Suominen, H., Ginter, F., Pysysalo, S., Airola, A., Pahikkala, T., Salanter, S., Salakoski, T.: Machine learning to automate the assignment of diagnosis codes to free-text radiology reports: a method description. In: Proceedings of the ICML/UIAI/COLT Workshop on Machine Learning for Health-Care Applications (2008)

38. Goksuluk, D., Korkmaz, S., Zararsiz, G., Karasagaloglu, A.E.: easyroc: an interactive web-tool for roc curve analysis using r language environment. RJ 8(2), 213–230 (2016)

39. Acharya, U.R., Oh, S.L., Hagivara, Y., Tan, J.H., Adam, M., Gertych, A., San Tan, R.: A deep convolutional neural network model to classify heartbeats. Computers in biology and medicine 89, 389–396 (2017)