Classification of Cardiac Arrhythmias in Photoplethysmographic Signals

Alberto Pettenella, *Giorgia Monzio Compagnoni*, and *Mohammadreza Jahednia*

M.Sc. Biomedical Engineering, AI in Biomedicine at Politecnico di Milano

Academic Year 2023/2024

1 Introduction

Premature atrial contractions (PACs) and premature ventricular contractions (PVCs) are commonly encountered cardiac arrhythmias whose physiological impact is still under investigation[1]. Although electrocardiograms represent the standard instrumentation to identify these arrhythmias, thanks to their distinctive waveform morphologies[2], the widespread diffusion of wearable devices, allowing for the continuous recording of photoplethysmographic (PPG) signals, may contribute to gathering more conclusive evidence regarding the malignant or benign nature of PACs and PVCs, ultimately improving diagnostic accuracy and informed clinical decision-making.

The goal of this study was thus to propose two novel detection approaches for PPG, discriminating between Normal/Abnormal beats and Normal/PAC/PVC beats, relying on hand-crafted as well as data-driven features extracted through deep learning models.

2 Materials and Methods

2.1 Data

The dataset used for the task is constituted by 105 patients, each provided with 30 minutes PPG recordings, the beat peak positions and their corresponding labels ("N" for normal, "S" for PAC and "V" for PVC). From the exploratory analysis three main findings surfaced:

- The signals exhibit non-uniform sampling frequencies, with 62 instances recorded at 128 Hz and 43 at 250 Hz.
- Class distribution is largely unbalanced with 93% 'N' beats, 4% 'S' beats and 3% 'V' beats.
- Each recording is affected by high frequency noise in given portions possibly due to subject's motion.

2.2 Preprocessing pipeline

Two preprocessing pipelines have been explored in the analysis mainly differing in the way artifacts are rejected. The adopted workflow (Fig. 1) is here discussed.

1. **Removal of signals with solely 'N' labels**. To avoid redundancy and slightly reduce class imbalance, 14 patients featuring only 'N' beats, all coming from the 250 Hz instances, were discarded from the dataset.
2. **Downsampling 250 Hz recordings to 128 Hz**. To ensure a consistent temporal representation of data, a downsampling approach was preferred over upsampling for three main reasons: higher expected computational efficiency, lower RAM memory usage and higher proportion of signals being at 128 Hz.

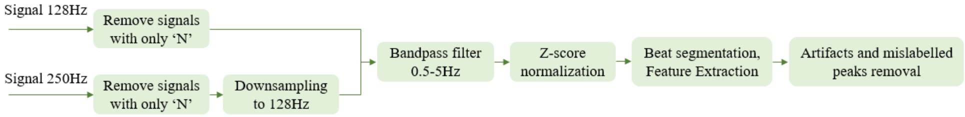
Signal 128Hz
Remove signals with only 'N'
Signal 250Hz
Remove signals with only 'N'
Downsampling to 128Hz
Bandpass filter 0.5-5Hz
Z-score normalization
Beat segmentation, Feature Extraction
Artifacts and mislabelled peaks removal

Figure 1: First preprocessing pipeline proposed.

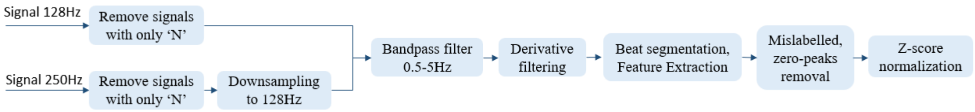
Signal 128Hz
Remove signals with only 'N'
Signal 250Hz
Remove signals with only 'N'
Downsampling to 128Hz
Bandpass filter 0.5-5Hz
Derivative filtering
Beat segmentation, Feature Extraction
Mislabelled, zero-peaks removal
Z-score normalization

Figure 2: Second preprocessing pipeline proposed.

3. **Bandpass filtering 0.5-5 Hz**. From the signal periodograms, it was observed that the frequency content is mostly contained between 0.5 and 3 Hz, ultimately a 5 Hz cutoff was chosen as it allows to better preserve the dicrotic notch where present.

4. **Z-score normalization** of each signal with respect to its own mean and standard deviation, to allow for inter-patient beat comparison while maintaining the individual patient's variability.

5. **Beat Segmentation and Feature Extraction**. Details on the procedure ensue in the following paragraph.

6. **Artifact and mislabelled peaks removal**. Once the segmented beats are obtained, artifact removal is performed by thresholding their amplitude. The threshold value is set empirically to 1.5. Subsequently, beats whose peak label was not found within a 20 points window centered on the actual peak were also eliminated, as their label was considered unreliable.

   The alternative approach to artifact removal (Fig. 2) was based on the signals' derivative: sections of the signal exhibiting a derivative exceeding a predefined threshold, empirically set at 0.4, were designated as noise and assigned a value of 0. If two noisy segments were identified within 50 or fewer samples of each other, the entire signal between them was classified as noisy and also set to 0.

Before actually discarding beats labeled as 'S' and 'V' based on the previous criteria, a reconstruction was attempted via autoencoders, yet results were unsatisfactory. Other approaches were thus considered to reduce class imbalance, namely downsampling of the majority class and minority class upsampling via SMOTE [3] and GANs [4]. Eventually, neither technique was adopted, only class weights were used during training instead.

Upon completing this phase, patients were partitioned into training, validation, and test sets. This allocation ensured that beats from the same patient were grouped together, while concurrently maintaining consistent class proportions throughout the partitioned datasets: 91% 'N' beats, 5% 'S' beats and 4% 'V' beats.

2.2.1 Beat Segmentation and Feature Extraction

Two types of segmentation were pursued. The first aims to partition the signal into individual beats, the second instead retains for each beat the previous and the following one. Both were implemented using a fixed-window approach of 100 and 200 points respectively, and a dynamic window approach, where the window size is taken as the average of the pre and post peak-to-peak distances in the single-beats segmentation and the double of such quantity in the contiguous-beats segmentation. A fixed-size window prevents the need for slicing or padding when feeding the beat as input to a model, however the dynamic-window is better tailored to the individual waveforms. For this reason, feature extraction was performed on the dynamically-partitioned single-beats. The features were computed based on [5] [6] [7] and they can be split into statistical, temporal, morphological and frequency-based attributes as in Tab. 1. Note that for the morphological characteristics, the dicrotic notch is not taken as a fiducial point as it was not present in all the beats.

Statistical	**Morphological**
Mean	Root Mean Square (RMS)
Standard Deviation	Energy
Skewness	Area
Difference in Skewness	Difference in Peak Values
Kurtosis	Difference in Minima Values
Difference in Kurtosis	Amplitude
Local Heart Rate Variability	Full Width at Half Maximum (FWHM)
Sample Entropy	Peak Value
Temporal	**Frequency-based**
Duration	Dominant Frequency
Rise Time	
Fall Time	
Peak-to-Peak Distance	
Local Peak-to-Peak Distance	

Table 1: List of Features

Correlation among features was assessed to remove any redundancy in the data representation. For feature pairs exhibiting a Pearson correlation higher than 0.9, the attribute with the highest average correlation to all the others was excluded from subsequent analysis. The redundant characteristics identified were: *mean, std, amplitude, peak value, duration and RMS.*

Subsequently, features were normalized across the datasets considering minima and maxima observed in training data to prevent any data leakage.

2.3 Normal/Abnormal Classification

2.3.1 Simple Feed Forward Neural Network

A shallow FFNN was developed taking as input the concatenation of individual beats with their respective features. The model is composed by three hidden layers with 64, 32 and 16 units respectively featuring *relu* activation function, while the output layer is made of 2 units to handle binary classification, for a total of 10,386 trainable parameters.

2.3.2 Siamese Networks

Siamese networks consists in two identical neural network architectures sharing the same weights. Their design focuses on learning meaningful representations by measuring distances in a low-dimensional embedding space, allowing the network to distinguish between similar and dissimilar instances.

In the analysis, the implementation of a siamese network required to randomly pair single beats and to assign each pair a new label: 0 for dissimilarity and 1 for similarity. Each beat pair was then fed to a Bidirectional Long Short Term Memory (LSTM) model and a compressed feature representation was obtained for every beat. Distances between the embeddings were then computed using the Euclidean metric and the contrastive loss function was used during training to maximize the distances between dissimilar elements and minimize the ones of similar examples.

At inference time, a template beat for class N, S and V was obtained by taking the median across each time point for each train beat, the templates were then duplicated to match the test set's length, predictions were performed and labels were assigned based on the lowest computed distance.

Templates allowed for higher computational efficiency compared to pairing each test set beat to a given number of normal and abnormal beats coming from the train set. Moreover, visualizing templates ensured the effectiveness of the segmentation and artifact removal procedures.

2.3.3 Anomaly detection via Autoencoder

Taking inspiration from [8], the dataset has been partitioned into distinct subsets: *normal beats*, comprising peaks labeled as 'N,' and *abnormal beats*, encompassing both 'S' and 'V' labeled peaks. All abnormal beats are used in the test set, while normal beats are split into train, validation and test sets. Being the autoencoder trained only on normal beats, it was expected that their reconstruction error at test time would be lower compared to the one of abnormal beats. Given this assumption, subsequent to model training, the Receiver Operating Characteristic (ROC) curve was generated by systematically varying thresholds to delineate beat classifications. The optimal threshold was subsequently identified employing two distinct methodologies:

1. Maximization of the penalized sensitivity metric (Par. 2.5).

2. Maximization of the difference between True Positive Rate (TPR) and False Positive Rate (FPR).

The final structure of the autoencoder was composed by 3 LSTM layers, with ReLu activation function, in both the encoder and the decoder part; dropout layers (dropout rate=0.2) have been introduced in between LSTM.

2.4 Normal/PAC/PVC Classification

Initial attempts to tackle the multiclass classification problem involved using four types of Deep Learning architectures: VGG-like, ResNet-like, Attention-based and LSTM-based. Each model was assessed both using only deep and the combination of deep and engineered features. In the latter case, the extracted features were concatenated to the latent representation of the beat and the newly formed feature vector was fed to a fully connected classification head. Nevertheless, results showed 'S' and 'V' beats commonly being mispredicted and rarely achieving an average sensitivity for the two minority classes above 0.5.

Consequently, a Machine Learning pipeline was followed taking advantage only of the handcrafted features: each model's hyperparameters were optimized via Bayesian search with a stratified cross validation scheme over five folds. Once the best hyperparameters had been found, according to the custom metric (Par. 2.5), the model was retrained with the same cross validation scheme and its performance was measured on train and validation sets to assess overfitting. The models tested were: RandomForest, Adaboost, XGBoost, LightGBM and Support Vector Machines.

Further fitting attempts were performed limiting the feature space to only-morphological, only-statistical and only-temporal attributes, however a drop in the penalized sensitivity metric (Par. 2.5) was observed, suggesting that combining the features allows for a greater discriminating power.

2.5 Penalized Sensitivity Metric

Aiming for the models to be used in a diagnostic setting, a custom metric was developed to prioritize the detection of the arrhythmias for both classification problems and its functioning is concisely illustrated in Alg. 1 and Alg. 2. The metric is primarily intended for model selection and early stopping when dealing with Artificial Neural Networks and Deep Learning.

Algorithm 1 Two-Class Scoring Metric

```
Require: predictions, labels, threshold, penalty
Compute sensitivity for class zero: $S[0]$
Compute sensitivity for class one: $S[1]$
if $S[0] < threshold$ then
    return $S[1] \times penalty$
else
    return $S[1]$
end if
```

In the case of Machine Learning models, the metric can easily be implemented as the strategy to evaluate the performance of cross-validated models when training or performing hyperparameter tuning.

In the analysis, a comparison between the *Penalized Sensitivity* and the *Macro Sensitivity*, when running Bayesian Search, yielded similar results, yet the custom metric led to a lower discrepancy between train and test set scores.

Algorithm 2 Three-Class Scoring Metric

```
Require: predictions, labels, threshold, penalty
    Compute sensitivity for class zero: $S[0]$
    Compute sensitivity for class one: $S[1]$
    Compute sensitivity for class two: $S[2]$
    Compute mean sensitivity for $S[1]$ and $S[2]$: $M$
    if $S[0] < threshold$ then
        return $M \times penalty$
    else
        return $M$
    end if
```

2.6 Confidence assessment

2.6.1 Binary Classifier

FFNN Since the output layer of the FFNN features a *Softmax* activation function, the model's confidence has been considered as the probability score for the predicted class.

Siamese Network Being the predictions of the network based on the distances between the learned representations, the confidence score is formulated as $1 - abs(d_{ij})$, where d_{ij} represents the distance between the $i - th$ beat's embedding and the $j - th$ template's embedding.

Autoencoder The confidence assessment for autoencoder predictions relies on the evaluation of each prediction error against the selected threshold (fig. 3). In particular, the confidence for a given prediction is taken as the normalized distance of the relative reconstruction error to the threshold.

2.6.2 Three Class Classifier

Random Forest Being a bagging algorithm, the prediction confidence is taken as the proportion of Decision Trees in the forest voting for the selected class as shown in fig. 4.

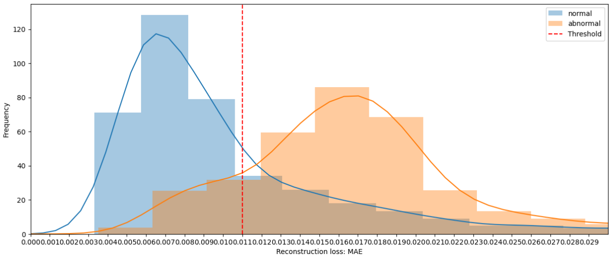

Figure 3: Autoencoder's predictions distribution.

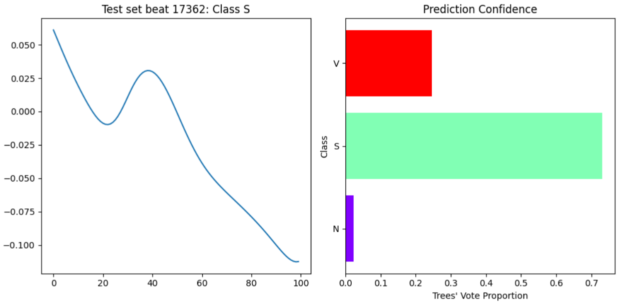

Figure 4: Confidence reported for a test beat.

Support Vector Machine Similarly to the autoencoder and the siamese network, the confidence score is distance-based: the prediction's certainty is given by the normalized distance to the discriminating boundary.

Boosting methods Confidence scores are taken as the predicted class probabilities of the input sample, computed as the weighted mean predicted class probabilities of the classifiers in the ensemble.

3 Results and Discussion

3.1 Normal/Abnormal Classification

Models' assessment has been conducted on the test set utilizing the macro-averages reported in tab. 2 together with the penalized sensitivity.

To our surprise, the FFNN achieved the best performance despite being the smallest and simplest architecture developed for the task. Nevertheless, when

Model	**Precision**	**Recall**	**F-1 score**	**Custom metric**
FFNN	**0.87**	**0.93**	**0.89**	**0.93**
Autoencoder	0.77	0.77	0.77	0.59
Siamese	0.70	0.84	0.74	0.71

Table 2: Binary classification results

the same network was applied to N/S/V classification, results were far from optimal, remarking the different tasks' complexities.

Enhancing the number of beat pairs could potentially improve the Siamese Network's ability to discern differences between beats. However, even a pool of 100 train beats, equally split into normal and abnormal, per test beat, resulted in lengthy and impractical inference times.

Interestingly, the autoencoder results were highly dependent on the weight initialization, making it the least stable solution.

3.2 Normal/PAC/PVC Classification

Macro-averages and the custom metric for test set examples are reported in tab. 3.

Model	**Precision**	**Recall**	**F-1 score**	**Custom metric**
Random Forest	**0.61**	0.72	**0.65**	0.615
SVM	0.56	0.69	0.60	0.58
Adaboost	0.5	0.64	0.53	0.462
XGBoost	0.63	0.67	0.65	0.53
LGBM	**0.61**	**0.73**	**0.65**	**0.635**

Table 3: Three class classification results.

Results show LGBM having the best performance, immediately followed by the Random Forest, despite being the latter more prone to overfit the train data.

Noticeably, all the models retained the ability to discriminate between Normal and Abnormal beats, mispredictions were instead common between PAC and PVC classes. A higher sensitivity for PAC, likely due to its greater proportion than PVC, was observed across all models except for Adaboost.

Although not explored in the analysis, we believe that a substantial improvement in N/S/V discrimination may be attained by training a model with the sole purpose of discriminating S and V beats and to run it on the abnormally predicted beats by the FFNN. In this way, we envision the identification of the most distinctive attributes between the two classes among the extracted ones.

4 Appendix

1D U-Net As an alternative to traditional approaches of classifying PPG beats, a 1D U-net architecture was implemented based on [9]. The input consisted into 30 seconds sequences whose labels are given by segments of masked beats, where masks are fixed-length and constant-amplitude beat representations. Amplitude serves as the beat encoding: 1 for N, 2 for S and 3 for V, while the 0 amplitude is representing inter-beat intervals, the background class. The model is therefore trained to return as output the beat masks for each sequence.

Following the reference paper, masks were not built considering the maxima of the softmax activation, but post-processing based on thresholds was implemented. This allowed also to address class imbalance by choosing a lower threshold on the minority classes.

Unfortunately, the approach did not lead to significant results, yet output masks showed that the occurrence of beats was recognized throughout sequences.

Artifacts management Existence of artifacts in the PPG signal is inevitable due to recording methods and devices. Our analysis has been entirely conducted under the hypothesis that no class information would be preserved in the beats belonging to the artifacts. However, once the best performing models were identified for both tasks, we attempted to train them using solely artifacts, for testing purposes. Surprisingly, results proved better than random choice, though still not comparable to the ones obtained on non-noisy beats.

To explain this result, we computed the templates (Par. 2.3.2) for N/S/V beats in the noisy portions and found a morphological similarity between them and the corresponding non-noisy beats.

References

[1] Jianyuan Hong, Hua-Jung Li, Chung chi Yang, Chih-Lu Han, and Jui chien Hsieh. A clinical study on atrial fibrillation, premature ventricular contraction, and premature atrial contraction screening based on an ecg deep learning model. *Applied Soft Computing*, 126:109213, 2022.

[2] Michele Orini, Stefan van Duijvenboden, William J Young, Julia Ramírez, Aled R Jones, Andrew Tinker, Patricia B Munroe, and Pier D Lambiase. Premature atrial and ventricular contractions detected on wearable-format electrocardiograms and prediction of cardiovascular events. *European heart journal. Digital health*, 4(2):112–118, 2023.

[3] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, Jun 2002.

[4] Alexander Nikitin, Letizia Iannucci, and Samuel Kaski. Tsgm: A flexible framework for generative modeling of synthetic time series. *arXiv preprint arXiv:2305.11567*, 2023.

[5] Elyas Sabeti, Narathip Reamaroon, Michael Mathis, Jonathan Gryak, Michael Sjoding, and Kayvan Najarian. Signal quality measure for pulsatile physiological signals using morphological features: Applications in reliability measure for pulse oximetry. *Informatics in Medicine Unlocked*, 16:100222, 2019.

[6] César A Millán, Nathalia A Girón, and Diego M Lopez. Analysis of relevant features from photoplethysmographic signals for atrial fibrillation classification. *International Journal of Environmental Research and Public Health*, 17(2):498, Jan 2020.

[7] Nathalia A Girón, César A Millán, and Diego M López. Systematic review on features extracted from ppg signals for the detection of atrial fibrillation. *Studies in Health Technology and Informatics*, 261:266–273, 2019.

[8] Moumita Roy, Sukanta Majumder, Anindya Halder, and Utpal Biswas. Ecg-net: A deep lstm autoencoder for detecting anomalous ecg. *Engineering Applications of Artificial Intelligence*, 124:106484, 2023.

[9] Dimitri Kraft, Gerald Bieber, Peter Jokisch, and Peter Rumm. End-to-end premature ventricular contraction detection using deep neural networks. *Sensors*, 23(20), 2023.