Scientific Paper / Artículo Científico

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR
INGENIUS
Revista de Ciencia y Tecnología
https://doi.org/10.17163/ings.n27.2022.04
pISSN: 1390-650X / eISSN: 1390-860X

A Deep Learning Approach to Estimate the Respiratory Rate from Photoplethysmogram

Un enfoque de aprendizaje profundo para estimar la frecuencia respiratoria del fotopletismograma

Lucas C. Lampier[1,*], Yves L. Coelho[1], Eliete M. O. Caldeira[1], Teodiano F. Bastos-Filho[1]

Received: 15-05-2021, Received after review: 18-08-2021, Accepted: 20-09-2021, Published: 01-01-2022

Abstract

This article describes the methodology used to train and test a Deep Neural Network (DNN) with Photoplethysmography (PPG) data performing a regression task to estimate the Respiratory Rate (RR). The DNN architecture is based on a model used to infer the heart rate (HR) from noisy PPG signals, which is optimized to the RR problem using genetic optimization. Two open-access datasets were used in the tests, the BIDMC and the CapnoBase. With the CapnoBase dataset, the DNN achieved a median error of 1.16 breaths/min, which is comparable with analytical methods in the literature, in which the best error found is 1.1 breaths/min (excluding the 8 % noisiest data). The BIDMC dataset seems to be more challenging, as the minimum median error of the literature's methods is 2.3 breaths/min (excluding 6 % of the noisiest data), and the DNN based approach achieved a median error of 1.52 breaths/min with the whole dataset.

Keywords: Deep Neural Networks, Photoplethysmography, Respiratory Rate

Resumen

Este trabajo presenta una metodología para entrenar y probar una red neuronal profunda (Deep Neural Network – DNN) con datos de fotopletismografía (Photoplethysmography – PPG), con la finalidad de llevar a cabo una tarea de regresión para estimar la frecuencia respiratoria (Respiratory Rate – RR). La arquitectura de la DNN se ha basado en un modelo de inferencia de frecuencia cardíaca (FC) a partir de señales PPG ruidosas. Dicho modelo se ha optimizado a través de algoritmos genéticos. En las pruebas realizadas se han utilizado dos conjuntos de datos de acceso abierto (BIDMC y CapnoBase). Con CapnoBase, la DNN ha logrado un error mediano de 1,16 respiraciones/min, que es comparable con los métodos analíticos en la literatura, donde el mejor error es 1,1 respiraciones/min (excluyendo el 8 % de datos más ruidosos). Por otro lado, el conjunto de datos BIDMC aparenta ser más desafiante, ya que el error mediano mínimo de los métodos de la literatura es de 2,3 respiraciones/min (excluyendo el 6 % de datos más ruidosos). Para este conjunto de datos la DNN ha logrado un error mediano de 1,52 respiraciones/min

Palabras clave: Redes Neuronales Profundas, Foto pletismografía, Frecuencia Respiratoria

[1,*]Postgraduate Program in Electrical Engineering, Universidade Federal do Espírito Santo (UFES), Vitória-ES, Brazil
Corresponding author ✉: lucas.lampier@hotmail.com.

Suggested citation: Lampier, L. C.; Coelho, Y. L.; Caldeira, E. M. O. and Bastos-Filho, T. F. (2022). «A Deep Learning Approach to Estimate the Respiratory Rate from Photoplethysmogram». Ingenius. N.° 27, (january-june). pp. 46-54. DOI: https://doi.org/10.17163/ings.n27.2022.04.

1. Introduction

Respiratory Rate (RR) is an important indicator of a person's physiological state, useful mainly to monitor pulmonary diseases. This physiological signal is traditionally measured by spirometry, pneumography or electromyography [1]. These methodologies are in general expensive and used mostly in medical environments. Photoplethysmography (PPG) is a cardiac signal usually measured using an oximeter, which is way cheaper than the traditional RR measurement methods, and recent studies have shown that it can be captured even remotely using a smartphone camera [2]. As respiration has an influence on the heart cycles, the RR information is also presented in the PPG signal [3].

Several methodologies in the literature had success on estimating the RR with a small error from the PPG signal [1], [4–8], however, attenuating noise is still a challenge to get robust predictions on low-quality PPG signals. This paper presents an alternative way to infer the Respiration Rate from PPG signals. Deep Learning techniques have been widely used in image problems, however it's also a powerful tool in one dimensional (1D) problems, mainly limited by the size and quality of the dataset, and also by the computer power in the training process. As the quantity of data available online growing, as well as the hardware computing power, DNNs are being tried in a large variety of problems that before were dominated by analytical procedures.

Regarding the PPG processing, in [1], the authors showed that a DNN can be used to extract the RR signal from PPG signals. And in [9], the pulse rate is extracted from the PPG using a DNN.

This work proposes to use a DNN to extract reliable RR measurements from PPG signals, by using analytical approaches to extract secondary signals from PPG and using DNN to infer the RR.

1.1. Related Work

According to [10] the PPG-RR algorithms can be summarized in five steps:

- Respiratory signal(s) extraction: This step consists of finding variation in the PPG signal related to the respiration cycles.
- Fusion of respiratory signals: The different signals extracted can be combined to create a unique signal with greater noise robustness (optional).
- Estimate the RR from a window: A window of the generated signal is segmented and the RR is estimated from it.
- Combine estimations: The result of different windows may be used to generate a final estimation (optional).
- Quality filtering: A quality score may be assigned to the PPG window to exclude low-quality predictions (optional).

Some of the first approaches that estimate RR from PPG use a highpass and a lowpass second-order Butterworth filters with cutoff frequencies at 0.1 Hz and 5 Hz respectively to remove noise from the PPG. And then the respiratory signal is estimated by applying a 0.4 Hz lowpass filter on the PPG signal [11]. The authors from [12] applied Singular Value Ratio (SVR) to extract the respiratory periodicity from the PPG, and then, they used Principal Component Analysis (PCA) to estimate the respiratory activity from the first principal component. In another approach, proposed by [13], they captured a PPG signal using the smartphone camera, and then the RR was estimated by finding the frequency corresponding to the highest peak on the spectrum generated by the Welch periodogram [7].

In [4], five different methodologies were compared (including their own). The oldest one was proposed in [14], and consists in apply a 16^{th} degree Bessel bandpass filter with cutoff frequencies of 0.13 Hz and 0.48 Hz (7.8 to 28.8 breaths/min). The second was proposed by [8], which consists in use a Fourier transform on the PPG signal and get the frequency with the highest amplitude in the bandwidth from 0.08 Hz to 0.4 Hz. In the methodology proposed by [5], auto-regressive (AR) models estimate a filter using the PPG signal, and, according to them, the frequency of the highest magnitude pole, inside the range of the RR bandwidth (0.08 Hz to 0.7 Hz), corresponds to the RR. Instead of using the PPG itself, in [6] three RR-related temporal features are estimated from it: Respiratory-Induced Intensity Variation (RIIV), Respiratory-Induced Amplitude Variation (RIAV), and Respiratory-Induced Frequency Variation (RIFV). Then, the power spectrum of each one is calculated and the RR is estimated by each one by taking the highest amplitude frequency in the RR frequency-band (0.067 Hz – 1.08 Hz) and calculating the mean of the three frequencies. Finally, the methodology proposed by [4] combines the AR modeling, proposed in [5] with the fusion of results of the three features proposed by [6].

The results presented in [4] show that for the CapnoBase dataset [15] the method that presented the lowest error was the method from [6] with a Median Absolute Error (MdAE) of 0.8 breaths/min using a 62 s window, however, 46 % of the PPG data were excluded due to noise. The second best in this dataset was the method from [5], which achieved a MdAE of 1.1 breaths/min and kept 92 % of the data. The BIDMC dataset appears to be more challenging, since the best MdAE was achieved by the method proposed by [8] (2.3 breaths/min), which was also one of the methods that kept most of the data (94 %).

A recent work has achieved even more accurate results [16], in which a method derives multiple waveforms from the PPG signal, and their quality are measured and then used as a weight to combine them using a Kalman Smoother. With this approach they achieved a median error of 0.2 breaths/min.

1.2. Deep Neural Networks (DNN)

According to [17], Deep Learning is an approach that can handle an important step in a machine layer problem: the features extraction. Its concept consists of concatenating multiple layers of simple models, where each model "learns" part of the problem-concepts, and so, a complex problem can be split into simpler ones to be better solved. An interesting point of this approach is that it performs the machine learning (classification, regression, clustering, etc.) and also the feature extraction part of the problem, in contrast with shallow techniques, in which the feature extraction step has to be done separately, in a previous stage, and demands domain knowledge. The limitation of the Deep Learning approach is the dataset. Normally, Deep learning models have a large number of weights to be adjusted, so, it needs a large amount of data to optimize them.

There are already some approaches that implemented Deep Learning techniques to solve problems related to physiological data. In [9], a DNN was implemented to estimate the Heart Rate (HR) having a PPG signal from a sensor worn on the user's wrist as input data. The model was tested with a challenging dataset, in which, the PPG signal was captured during various physical activities, which inserts noise and artifacts on it. The architecture of the model consisted of two 1-D convolutional layers followed by two Long Short-Term Memory (LSTM) layers and finally a dense layer at the end, and it reached a Mean Absolute Error (MAE) of 1.47 $\pm$ 3.37 Beats Per Minute (BPM).

A Deep Learning approach to extract the RR signal from the PPG is presented in [1]. Their model consists of several convolutional layers connected to several deconvolutional layers that transform the PPG signal into a respiratory signal, and then they calculate the RR from it. Their method was tested in two independent datasets, the CapnoBase and the Vortal [18] and reached a Mean Squared Error (MSE) of 0.262 in the first dataset and an error of 0.145 in the second one (which corresponds to a Root Mean Squared Error (RMSE) of approximately 3.1 breaths/min and 2.3 breaths/min respectively).

2. Materials and methods

2.1. Datasets

Two datasets were used in the tests: the CapnoBase benchmark dataset [15] and the BIDMC PPG and Respiration Dataset [19]. The CapnoBase benchmark dataset [20], contains 42 recordings, with 8 min each, containing the PPG and the inhaled and exhaled carbon-dioxide (CO_2) signal, both collected with a sampling frequency of 300 Hz. The BIDMC PPG and Respiration Dataset [21], contains 8 min PPG and CO_2 breath signal recordings from 53 volunteers collected at 125 Hz.

2.2. Photoplethysmogram Respiratory Features

Three signals related to the respiration were generated from PPG [6]:

1. Respiratory-Induced Intensity Variation (RIIV): As the amplitude of the PPG signal varies in synchrony with the respiratory cycle, this feature can be used to estimate the RR [3]. This series is estimated by taking the intensity value of each peak in the PPG signal [4].

2. Respiratory-Induced Amplitude Variation (RIAV): The breath-cycles induce an amplitude-variation in the PPG waves, which can be estimated by calculating the variation in the peak-valley difference in the PPG waves [6].

3. Respiratory-Induced Frequency Variation (RIFV): The HR variation is also highly linked to the inspiration and expiration events of breathing. The inspirations increase the HR and the expiration decreases it. This signal may be represented by the time interval between consecutive PPG peaks [6].

A graphic representation of the RIIV, RIAV and RIFV features is presented in Figure 1.

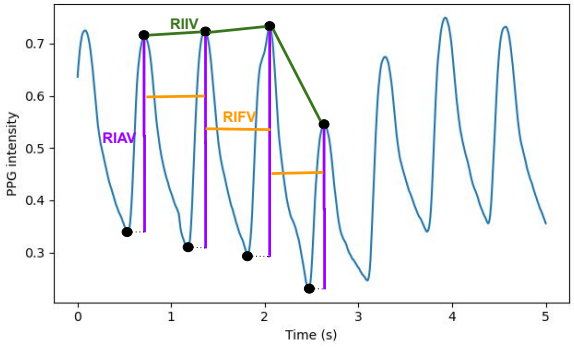

Figure 1. Extracted RR-related PPG features

2.3. Signal Quality Index (SQI)

The SQI quantifies the noise and artifact contamination in the analyzed PPG window. This index is used to exclude signals that do not reach a minimum

quality. In this work, the SQI is calculated targeting two points: "flat-lines" in the signal, and concordance between different peak detectors [4]. The "flat-lines" are defined as part of the PPG signal in which the difference between consecutive samples is near to zero. To detect it, the following method was performed:

1. Calculate the $StdDiff$ signal as follows:

   $StdDiff[i-wdw] = \sigma(diff(PPG[i-wdw:i])),$

   $for\ i = wdw, wdw+1, wdw+2, ..., N.$

   Where $\sigma(.)$ is the standard deviation operation, $diff(.)$ is the first derivative operation, PPG is the PPG signal, wdw is the window length, that, in the case, is the integer part of *1.5 s* $\times$ *frequency sample* (1.5 s is the time interval between consecutive peaks on the minimum HR value, which is normally 40 BPM), and N is the number of samples in the PPG signal.

2. Initialize the Q array with ones and the same size as PPG: $Q = ones(N)$

3. Calculate the threshold value: $th = \mu(StdDiff) - \alpha \times \sigma(StdDiff))$, where $\mu(.)$ is the mean operation and α is an adjustable parameter.

4. Set the values of k to zero in the corresponding locations where $StdDiff$ is too low:

   $for\ i = wdw, wdw+1, wdw+2.....N : \{$

   $\quad if \quad (StdDiff[i-wdw] < (th) : \{$

   $\quad\quad Q[i+wdw] = 0$

   $\}\}$

5. The value of K is the proportion of one values in Q inside the analyzed PPG window, over the total size of it.

The concordance between the two peak-detectors was performed as following: if the difference between the position found by the detectors on the same peak is smaller than 150 ms, it is set that both detectors agree on that peak, therefore its position is correct, otherwise, the peak is considered a noise [4]. An F1-score is then applied to measure the concordance between the two peak-detectors on the analyzed segment of the signal. The peak detection methods were provided by two different *20* libraries: the *Neurokit2* [22], which implements the PPG peak detection algorithm provided in [23] and the *Heartpy* library [24], in which the authors implemented their own peak detection technique.

Finally the SQI is computed as: $SQI = F1 \times K$, where $F1$ is the concordance between peaks and K is the "flat-line" proportion.

2.4. DNN Architecture

The architecture of the model was inspired on the CorNET [9], which has two convolution blocks, each containing a convolution layer (32 filters with a kernel size of 40), a batch normalization layer, a Rectified Linear Unit (ReLU) activation layer, and finally a max-pooling layer with a pool size of 4 (a Dropout layer with a drop rate of 0.1 was used for training). After the convolution blocks two LSTM layers are concatenated, each one with 128 units and the hyperbolic tangent ($tanh$) activation function. To predict the HR a single neuron with linear activation is used.

The architecture proposed to predict the RR is a result of genetic optimization on the CorNET's parameters presented in Table 1. There is always an additional dense layer with one neuron with linear activation at the end of the model.

Table 1. Parameters tried in the DNN architecture

Parameter	Possibilities
# of convolution blocks	0/1/2
# of convolution filters/layer	128/64/32
Length of convolution filters	11/21/31
# of LSTM layers	1/2/3
# of LSTM units/layer	128/64/32
# of dense layers	0/1/2
# of neurons/layer	128/64/32/16

The parameters above are optimized, while the non-cited parameters are used as defined by default in the *Keras* library [25]. As the number of combinations is very large (2916), the parameters are optimized via genetic optimization. This strategy starts by defining a population of L models with random parameters and then all models are trained, tested, and sorted by the result. The M models with the best performance are kept while the others are discarded. The kept models set is called *parents*, then $M - J$ new models are generated by taking values of three different models (one parent for the convolution parameters, one parent for the LSTM, and one for dense). Also, there is a chance of H % that the parameters of the new models change randomly (these changes are called mutations), these new models are called $offspring$. There is also generated J new completely random models. We give this set the name of $foreigners$. Then the *parents*, $offspring$ and $foreigners$ sets are united and a new interaction starts. The procedure is repeated for E epochs. The idea is that the parameters that decrease the estimation error will prevail in the *parents* set, and their combination may generate a model on the $offspring$ set that overcomes its parents' result. Also, new random parameters are inserted with the mutation and the $foreigners$ set at each iteration. These

random additions give a chance for the population of models to get out of a sub-optimal local minimum.

For this experiment, the parameters of the genetic optimization were: $L = 45$, $M = 20$, $J = 5$, $E = 20$ and $H = 20$, this reduced the number of trained models from 2916 to 520 (25 new models/epoch $\times$ 20 epochs + the first 20 parents). And the best model found is a DNN with 2 convolution blocks with 32 filters and one MaxPooling-1D (kernel size of 2) on each convolution layer, a kernel size of 21, and ReLU activation. Then, one LSTM layer with 32 neurons, and finally, 2 dense layers, one with 32 neurons and ReLU activation and the last one with one neuron and linear activation. A summary of the final DNN architecture is presented on Figure 2.

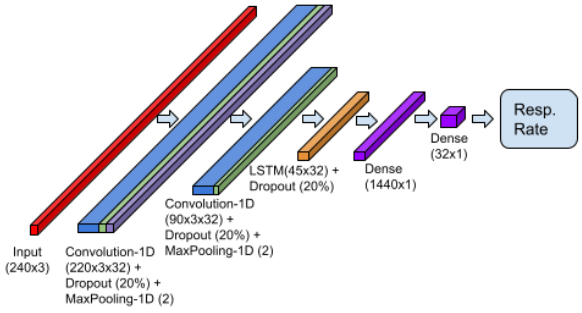

Figure 2. Selected Neural Network architecture

2.5. Training and Testing Procedure

To train and evaluate the model two datasets were used: the BIDMC [19] and the CapnoBase [15] datasets. The cross-validation method with four groups were used: for each database, their files were split into four groups, each group containing 25% of the data of each dataset. Then, the same procedure was performed on each group: first, the PPG peaks were found using the *Neurokit2* library [22]. Then using a sliding window of 64 s with step of 4 s, the three RR-related signals (RIIV, RIAV, and RIFV) were computed from it and then they were interpolated to an artificial sampling frequency of 4 Hz and the first 240 samples. As the peak will probably not start at the beginning and the end of the window, the interpolated signal may variate between 61 s to 64 s. (60 s) were extracted from the window generating a sample with 240 lines (temporal steps) and 3 columns (one for each: RIIV, RIAV, and RIFV). The final step is to normalize the data so each channel of the new sample has mean 0 and variance 1: $Y = (X - \mu(X))/\sigma(X)$, where Y is the normalized series, X the input series, $\mu(.)$ is the mean operation and $\sigma(.)$ is the standard deviation operation.

To get the reference value of RR, the peak of the CO_2 correspondent signal of each sample is located using the Respiration module of *Neurokit2* and then, the Median RR of the 64 s sample is calculated by taking the inverse of the median value of consecutive peaks time-interval.

The four groups of normalized samples are then split into the training and testing sets, where the training set has 3 groups and the test has 1. The training and testing procedures were repeated four times, so each group could be evaluated in the test set once. Each of the 4 models (one for each test group) was trained for 1400 epochs using the Adam optimizer with standard parameters from *Keras* [25]. The processing was made on a *Ryzen 5 3500X* computer with 16 GB of RAM memory and a *NVIDIA GeForce RTX 2060 SUPER* with 8 GB dedicated memory. The Mean Squared Error (MSE) metric was used to compute the loss for each epoch. The loss curve result is presented in Figure 3.

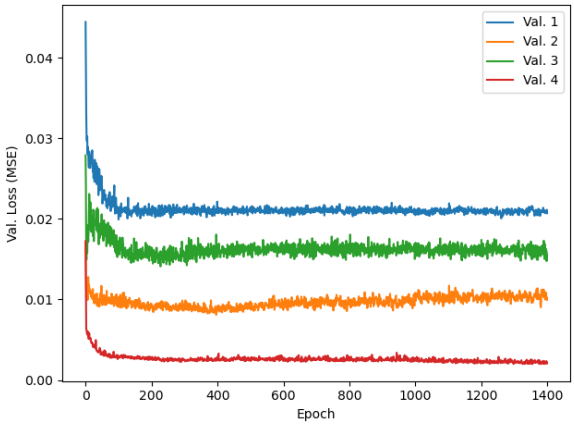

Figure 3. Ts curve of the 4 instances of the network

The validation groups 1 and 4 had a loss decreasing until epoch 1400, the validation group 2 has it's minimum loss near the epoch 400 and validation group 3 had a minimum around epoch 200, them it decreased again after epoch 800. The graph was used to determine the number of epochs to train the DNNs, as half of the Validation groups still decreasing until epoch 1400 and the dropout layers seems to prevent overfitting on the other groups, since the loss do not increase to much after the minimums. We repeated the process shuffling the data and the groups, and training 4 DNNs (one for each train-test group) for 1400 epochs to perform the cross validation.

3. Results and Discussion

The Median Absolute Error (MdAE) and the standard deviation of the error (STD) for each dataset are presented in Table 2 and Table 3, as well as are presented the Root Mean Squared Error (RMSE), and the SQI threshold used to filter noisy samples. By testing different SQI values, it is possible to analyze the influence of the signal quality on the result.

Table 2. Error measurements of the DNN based RR prediction

BIDMC dataset				
SQI	**N (%)**	**MdAE (bts/min*)**	**STD (bts/min)**	**RMSE (bts/min)**
0.90	87	1.39	6.55	6.59
0.93	80	1.39	6.57	6.62
0.97	66	1.26	6.51	6.55
1.00	37	1.04	4.47	4.47
0.00	100	1.52	6.86	6.94

* bts/min is equal to breaths/min

The MdAE results achieved by our model on the CapnoBase dataset are compatible with the ones from the best methodologies compared in [4], in which, the best MdAE for a 64 s window was 0.8 breaths/min, achieved using [6] method. However, the methodology excludes 36 % of samples that do not achieve the minimum SQI. The second best was the method from [5], achieving a MdAE of 1.1 breaths/min for 92 % of the best samples. For the same dataset, our method got a MdAE of 1.16 breaths/min (calculated concatenating the results of all four tests), and a MdAE of 1.11 breaths/min with 93% of the highest SQI samples. However, the RMSE of the proposed DNN model did not reach the error from the RespNET [1], which is 3.1 breaths/min, indicating that our model can still be refined to achieve better results.

Table 3. Error measurements of the DNN based RR prediction

CapnoBase dataset				
SQI	**N (%)**	**MdAE (bts/min*)**	**STD (bts/min)**	**RMSE (bts/min)**
0.90	98	1.16	5.93	5.94
0.93	97	1.16	5.94	5.95
0.97	93	1.11	5.65	5.66
1.00	73	1.06	4.46	4.97
0.00	100	1.16	5.88	5.90

* bts/min is equal to breaths/min

The proposed DNN model scored smaller errors on the BIDMC dataset than all other methods compared in [4]. With all samples, our method got a MdAE of 1.52 breaths/min in this dataset, while the smallest error found for this dataset in [4], was the methodology proposed by [8], with a MdAE of 2.3 breaths/min. This improvement in result shows the success in DNN methods to process physiological data. Where, with a dataset large enough and the right complexity, important information can be extracted from the data while noise and artifacts are discarded.

It is also notable in Table 2 and Table 3 that the MdAE by itself does not measure a good result. Besides having a low MdAE the results present a considerable RMSE, as it reinforces highly variation results and the median computation does not take into account how much the extreme results deviate from the desired value, only the middle value(s) is(are) computed. This can be observed in the scatter plot of the predictions and true values presented in Figure 4, and Figure 5.

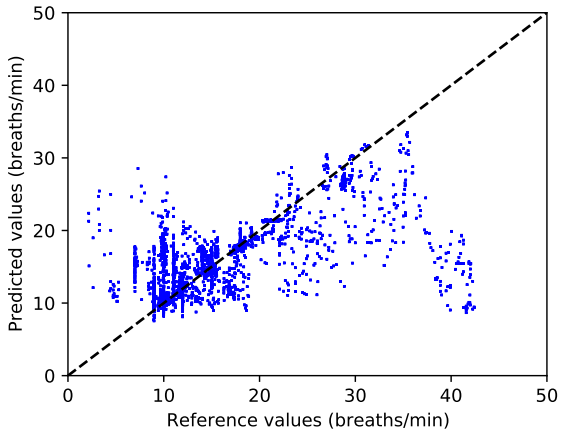

Figure 4. Scatter plot of the RR predictions and the true values of the CapnoBase dataset. The samples ploted have a SQI over 0.9

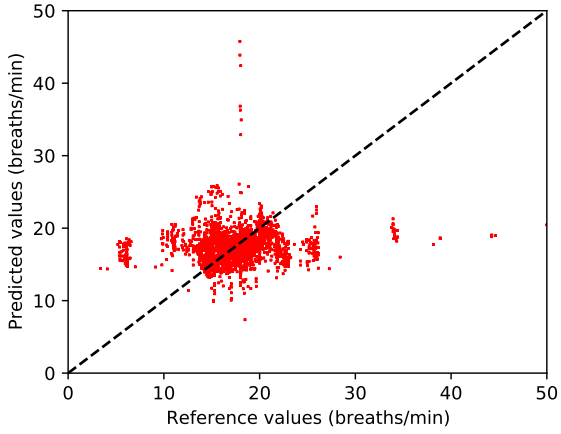

Figure 5. Scatter plot of the RR predictions and the true values of the BIDMC. The samples ploted have a SQI over 0.9

To better compare the performance of our method against the researched literature, a summary of the results of each method is presented in Table 4. The best result presented in the Capnobase dataset was the method proposed by [16], which reached a MdAE of 0.2 breaths/min, whereas the previous benchmark methods on the same dataset achieved a minimum MdAE of 0.8 breaths/min (excluding 36% of the noisiest data). Our method, using the whole data, achieved a MdAE of 1.2 breaths/min. The BIDMC database seems more challenging, as the methods tested on this dataset presented a minimum MdAE of 2.3 breaths/min, and our

method achieved a MdAE of 1.5 breaths/min. The authors from [16] did not used de BIDMC dataset in their study, so, we are not able to compare the results with their method in this dataset.

Table 4. Error measurements of different methods presented in the literature. N is the percentage of PPG windows used, median absolute error (MdAE) and 25^{th} and 75^{th} percentiles (25TH - 75TH) Root Mean Squared Error (RMSE) and Mean absolute error (MAE) and Standard deviation (STD)

	Dataset							
	Capnobase				**BIDMC**			
Method	N (%)	MdAE (25TH - 75TH)	RMSE	MAE (STD)	N (%)	MdAE (25TH - 75TH)	RMSE	MAE (STD)
Proposed	100	1.2 (0.4-3.4)	5.9	3.1 (5.0)	100	1.5 (0.6-3.6)	6.9	3.4 (6.0)
Khreis (2020)	100	0.2 (0.1-0.9)	-	-	-	-	-	-
Ravichandra (2019)	-	-	3.1	-	-	-	-	-
Pimentel (2017)	92	1.9 (0.3-3.4)	-	-	94	2.7 (1.5-5.3)	-	-
Shelley (2016)	92	2.2 (0.2-8.3)	-	-	94	2.3 (0.9-7.9)	-	-
Karlen (2013)	64	0.8 (0.3-2.7)	-	-	34	5.7 (1.5-9.7)	-	-
Fleming (2007)	92	1.1 (0.4-3.5)	-	-	94	5.5 (2.7-8.1)	-	-
Nilsson (2000)	92	10.2 (4.8-12.4)	-	-	94	4.6 (2.5-8.5)	-	-

4. Conclusions

This work presents a methodology to use a DNN approach to estimate RR using PPG signals. The model was inspired in a previous successful architecture, which was optimized to our problem using an adapted genetic optimization. The inputs to the DNN are three RR-related respiratory signals extracted from the PPG, and the output is the corresponding RR-value. To test the methodology the files of two open-access datasets were split into 4 folds, keeping the same proportion of each dataset files in each folder. A training-test procedure was repeated 4 times each one with 3 folds for training and 1 for testing, so every fold was used once as a test. The results achieved are comparable with the most of the benchmark analytical methodologies using the CapnoBase dataset, and the DNN overcame them in the more challenging BIDMC dataset, showing the success of the methodology to process physiological data.

Although successful, the performance of analytical methods is usually accompanied by heuristic thresholds or a large number of expertly-tuned free parameters, which could prevent generalization of the developed methodologies. However, DNN approaches are designed to generalize the data as much as possible and to have as less as preprocessing as possible.

Acknowledgements

The authors thank CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), an agency of the Brazilian Ministry of Science, Technology, Innovations and Communications that supports scientific and technological development, FAPES (Fundação de Amparo à Pesquisa e Inovação do Espírito Santo), an agency of the State of Espírito Santo, Brazil, that supports scientific and technological development, and CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for the financial support granted to this work.

References

[1] V. Ravichandran, B. Murugesan, V. Balakarthikeyan, K. Ram, S. P. Preejith, J. Joseph, and M. Sivaprakasam, "RespNet: A deep learning model for extraction of respiration from photoplethysmogram," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 5556–5559. [Online]. Available: https://doi.org/10.1109/EMBC.2019.8856301

[2] A. Floriano, L. Lampier, R. S. Rosa, E. Caldeira, and T. Bastos-Filho, "Remote vital sign mon-

itoring in accidents," *Polytechnica*, vol. 4, no. 1, pp. 26–32, Apr. 2021. [Online]. Available: https://doi.org/10.1007/s41050-020-00027-1

[3] D. J. Meredith, D. Clifton, P. Charlton, J. Brooks, C. W. Pugh, and L. Tarassenko, "Photoplethysmographic derivation of respiratory rate: a review of relevant physiology." *Journal of medical engineering & technology*, vol. 36, no. 1, pp. 1–7, jan 2012. [Online]. Available: https://doi.org/10.3109/03091902.2011.638965

[4] M. A. Pimentel, A. E. Johnson, P. H. Charlton, D. Birrenkott, P. J. Watkinson, L. Tarassenko, and D. A. Clifton, "Toward a robust estimation of respiratory rate from pulse oximeters," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 8, pp. 1914–1923, 2017. [Online]. Available: https://doi.org/10.1109/TBME.2016.2613124

[5] S. G. F. L. Tarassenko, "A comparison of signal processing techniques for the extraction of breathing rate from the photoplethysmogram," *International Journal of Biological and Medical Sciences*, vol. 1, no. 6, pp. 366–370, 2007. [Online]. Available: https://bit.ly/3Cprqqg

[6] W. Karlen, S. Raman, J. M. Ansermino, and G. A. Dumont, "Multiparameter respiratory rate estimation from the photoplethysmogram." *IEEE transactions on bio-medical engineering*, vol. 60, no. 7, pp. 1946–1953, jul 2013. [Online]. Available: https://doi.org/10.1109/TBME.2013.2246160

[7] L. Nilsson, T. Goscinski, S. Kalman, L. G. Lindberg, and A. Johansson, "Combined photoplethysmographic monitoring of respiration rate and pulse: A comparison between different measurement sites in spontaneously breathing subjects," *Acta Anaesthesiologica Scandinavica*, vol. 51, no. 9, pp. 1250–1257, 2007. [Online]. Available: https://doi.org/10.1111/j.1399-6576.2007.01375.x

[8] K. H. Shelley, A. A. Awad, R. G. Stout, and D. G. Silverman, "The use of joint time frequency analysis to quantify the effect of ventilation on the pulse oximeter waveform." *Journal of clinical monitoring and computing*, vol. 20, no. 2, pp. 81–87, apr 2006. [Online]. Available: https://doi.org/10.1007/s10877-006-9010-7

[9] D. Biswas, L. Everson, M. Liu, M. Panwar, B.-E. Verhoef, S. Patki, C. H. Kim, A. Acharyya, C. Van Hoof, M. Konijnenburg, and N. Van Helleputte, "Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 13, no. 2, pp. 282–291, 2019. [Online]. Available: https://doi.org/10.1109/TBCAS.2019.2892297

[10] P. H. Charlton, D. A. Birrenkott, T. Bonnici, M. A. F. Pimentel, A. E. W. Johnson, J. Alastruey, L. Tarassenko, P. J. Watkinson, R. Beale, and D. A. Clifton, "Breathing rate estimation from the electrocardiogram and photoplethysmogram: A review," *IEEE Reviews in Biomedical Engineering*, vol. 11, pp. 2–20, 2018. [Online]. Available: https://doi.org/10.1109/RBME.2017.2763681

[11] L. G. Lindberg, H. Ugnell, and P. A. Oberg, "Monitoring of respiratory and heart rates using a fibre-optic sensor," *Medical & Biological Engineering & Computing*, vol. 30, no. 5, pp. 533–537, 1992. [Online]. Available: https://doi.org/10.1007/BF02457833

[12] K. V. Madhav, M. R. Ram, E. H. Krishna, K. N. Reddy, and K. A. Reddy, "Estimation of respiratory rate from principal components of photoplethysmographic signals," *Proceedings of 2010 IEEE EMBS Conference on Biomedical Engineering and Sciences, IECBES 2010*, no. December, pp. 311–314, 2010. [Online]. Available: https://doi.org/10.1109/IECBES.2010.5742251

[13] Y. Nam, Y. Kong, B. Reyes, N. Reljin, and K. H. Chon, "Monitoring of heart and breathing rates using dual cameras on a smartphone," *PLoS ONE*, vol. 11, no. 3, mar 2016. [Online]. Available: https://doi.org/10.1371/journal.pone.0151013

[14] L. Nilsson, A. Johansson, and S. Kalman, "Monitoring of respiratory rate in postoperative care using a new photoplethysmographic technique," *Journal of Clinical Monitoring and Computing*, vol. 16, no. 4, pp. 309–315, 2000. [Online]. Available: https://doi.org/10.1023/A:1011424732717

[15] W. Karlen, "CapnoBase IEEE TBME Respiratory Rate Benchmark," 2021. [Online]. Available: https://doi.org/10.5683/SP2/NLB8IT

[16] S. Khreis, D. Ge, H. A. Rahman, and G. Carrault, "Breathing Rate Estimation Using Kalman Smoother with Electrocardiogram and Photoplethysmogram," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 3, pp. 893–904, 2020. [Online]. Available: https://doi.org/10.1109/TBME.2019.2923448

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org. [Online]. Available: https://bit.ly/3Eh4Twb

[18] P. H. Charlton, T. Bonnici, L. Tarassenko, D. A. Clifton, R. Beale, and P. J. Watkinson, "An

assessment of algorithms to estimate respiratory rate from the electrocardiogram and photoplethysmogram." *Physiological measurement*, vol. 37, no. 4, pp. 610–626, apr 2016. [Online]. Available: https://doi.org/10.1088/0967-3334/37/4/610

[19] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals." *Circulation*, vol. 101, no. 23, pp. e215–e220, jun 2000. [Online]. Available: https://doi.org/10.1161/01.cir.101.23.e215

[20] CapnoBase. (2020) Capnobase is a collaborative research project that provides an online database of respiratory signals and labels obtained from capnography, spirometry and pulse oximetry. [Online]. Available: https://bit.ly/3EjfHKm

[21] M. A. F. Pimentel, A. E. W. Johnson, P. H. Charlton, D. Birrenkott, P. J. Watkinson, L. Tarassenko, and D. A. Clifton, "Toward a robust estimation of respiratory rate from pulse oximeters," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 8, pp. 1914–1923, 2017. [Online]. Available: https://doi.org/10.1109/TBME.2016.2613124

[22] D. Makowski, T. Pham, Z. J. Lau, J. C. Brammer, F. Lespinasse, H. Pham, C. Schölzel, and S. H. A. Chen, "Neurokit2: A python toolbox for neurophysiological signal processing," *Behavior Research Methods*, vol. 53, no. 4, pp. 1689–1696, Feb 2021. [Online]. Available: https://doi.org/10.3758/s13428-020-01516-y

[23] M. Elgendi, I. Norton, M. Brearley, D. Abbott, and D. Schuurmans, "Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions," *PLOS ONE*, vol. 8, no. 10, pp. 1–11, 10 2013. [Online]. Available: https://doi.org/10.1371/journal.pone.0076585

[24] P. van Gent, H. Farah, N. van Nes, and B. van Arem, "Heartpy: A novel heart rate algorithm for the analysis of noisy signals," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 66, pp. 368–378, 2019. [Online]. Available: https://doi.org/10.1016/j.trf.2019.09.015

[25] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.