

Validation of Data correlation - Heart Rate and Respiratory Rate from ECG in Python

Stefan Jovanov, Bojan Ristovski, Ana Madevska Bogdanova, and Magdalena Kostoska

University Ss. Cyril and Methodius,
Faculty of Computer Science and Engineering,
Rugjer Boskovikj 16, 1000 Skopje, Macedonia
{stefan.jovanov,bojan.ristovski}@students.finki.ukim.mk
{ana.madevska.bogdanova,magdalena.kostoska}@finki.ukim.mk

Abstract. Wearable biosensors that measure ECG, used in time critical situations of mass casualties, when fast triage procedure must be accomplished, might not support heart rate (HR) and respiratory rate (RR) measurement. These parameters can be calculated using the ECG signal. In this paper we are exploring the use of Python libraries by validating the correlation between the HR and RR calculated from an ECG signal and the measured HR and RR for the same patients, at the same time. The correlation for HR is 0.9, but the RR correlation is very low. Another part of the paper elaborates the importance of the data quality.

Keywords: ECG, respiratory rate, heart rate, signal processing

1 Introduction

Deriving vital sign parameters at fast pace in the cases with mass casualties in combat or natural disaster events (floods, earthquakes, terrorist attacks) when the number of injured reaches hundreds of people, it is essential for the triage process to be performed time efficiently. Following the triage procedure [5], in order to obtain the medical status of an injured person almost instantly, one needs to use heart rate (HR) and respiratory rate (RR) vital signs. These parameters could be extracted from a given ECG signal obtained from a biosensor that does not support HR and RR measurements.

An electrocardiogram (ECG) is a medical test that detects heart electrical signals by measuring the electrical activity generated by the heart as it contracts. This is done by ECG machines, connected with electrodes to certain points of the patient's body, providing reading the information and get results about the person's heart condition. ECG is easily read from wearable biosensors as well. In the disaster events with mass casual occurrence, it is very important to relay on the fast calculated HR and RR derived from common low-cost ECG biosensor.

Another important aspect of designing models using Machine Learning methods, testing algorithms for Blood pressure or SPO2 estimation that uses ECG/HR/RR

or PPG signals, as well as in general algorithms regarding human body state and functioning, is the data quality. The operation of data collection, while strictly following defined procedure, can still yield incorrect reading due to sensor displacement or incorrect patient position during the measurement. Another relevant obstacles can be the perspiration or general humidity settings. This can result in missing values or inconsistencies readings that can occur any time during recording. Data quality is very important, since only correct data can provide correct state evaluation and confirm correctness of the research algorithm. Thus data preprocessing and quality evaluation is important in the early steps of the research.

In this paper we will analyze recorded data - raw ECG signals and the correlation between the calculated HR and RR (using Python libraries) and the measured HR and RR for the same patients recorded at the same time as the ECG signal. Another important goal is to detect errors and inconsistencies and evaluate the quality of the collected data.

The structure of this research paper is the following. Section II presents an overview on similar papers or related researches, and Section III elaborates the methods used for the experiment, specifically the processing and visualization of the ECG signals. The results of the experiment are presented in Section IV and performance evaluation is discussed in Section V. We conclude the paper in Section VI.

2 Related Work

Digital signal processing is a method that is used in a variety of biomedical researches consisted of processing ECG signals. Researchers can extract a lot of information from the recorded raw ECG signals, and this data can be analyzed using various algorithms. Most of those algorithms are based on the modulation of the QRS complex amplitude (the main spike of the signal)[3], the RSA (respiratory sinus arrhythmia, in close relation with the heart rate)[10][3] and the RPA (R-peak amplitude)[10].

The heart rate derived from the processed ECG signal is based on the QRS complex of the signal[12]. The processing of this signal depends on the noise of itself, but usually the devices used for measuring the signal are reliable, and determining the heart rate rarely poses a problem. This method is applied in calculating the Heart Rate Variability, which can be used for different goals, few of them being: evaluating cardiovascular risk stratification, evaluating sinoatrial node function, monitoring mental stress, and evaluating anesthesia outcome.[9]

The calculation of the respiratory rate from the ECG signal can be done by a lot of different methods, depending on the data that is needed. Such algorithms are: the previously mentioned RSA and RPA [10], PQRST complex(the pattern of electrical activity of the heart during one cardiac cycle)[13], the Fourier Transformation[2] etc. Even though these methods give good results, there are always certain samples that are going to give us errors, mainly due to the noise and other artifacts in the signal. In order to solve these problems, the signals need to be

preprocessed by an algorithm that is going to strengthen the signal and remove the noise [1]. The information provided by the Respiratory Rate is extremely important for evaluating the health situation of the patient[8]. For example, we can predict a non-sudden cardiac problems[14], difficulties maintain homeostatic control (the body’s internal environment), hypoxia (low levels of oxygen in the cells), hypercapnia (high levels of carbon dioxide in the bloodstream), metabolic and respiratory acidosis [1]. As mentioned in the Introduction, one needs to use heart rate and respiratory rate vital signs in order to evaluate the patients state in the Triage process [5].

3 Methods

The ECG, HR and RR were measured using the biosensor from the ZEPHYR™ Performance systems, Zephyr BioHarnes [15]. The protocol of data collection consisted of creating 30 seconds long recording in sitting position with 250 Hz sampling rate ECG recording of each patient, along with a measured Heart Rate and Respiratory Rate at the same period of time. The database consists of 117 recordings from 32 different patients.

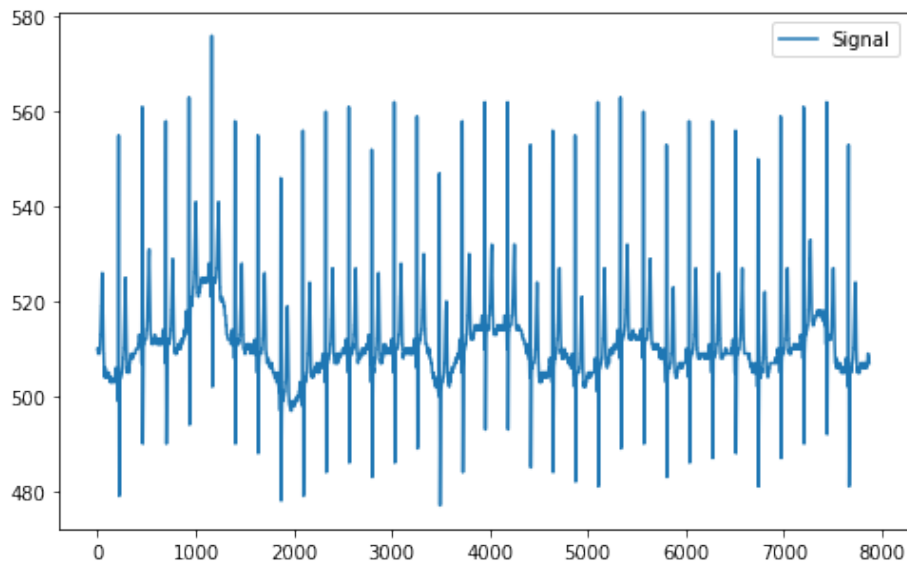


Fig. 1. Example visualization of an ECG signal.

In order to accomplish our study, we used the following Python libraries as a key aspect of our research: Heartpy [4, 11], Matplotlib [7] and Numpy [6] that provide many different functions. Numpy is mostly used to read and

manipulate the data, Heartpy is used to convert the data into ECG, heart rate and respiratory rate signals and Matplotlib is used to visualize the data.

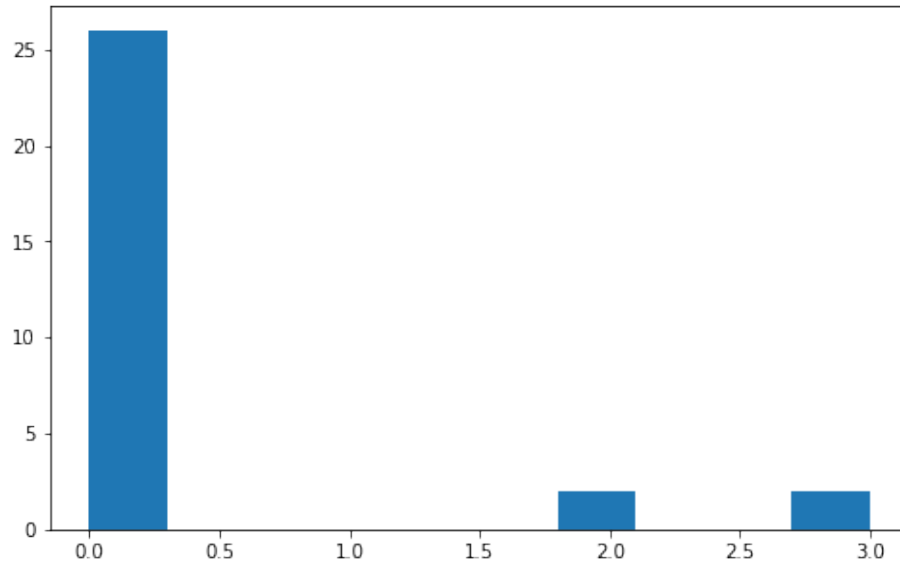


Fig. 2. Example of bad signal - histogram for respiratory rate.

3.1 Processing the signals

The recorded signals are processed to obtain an expected signal representation due to used sensor and its frequency as well as the accompanied software used. The processing is performed in several steps:

Step 1: first we read the data for the ECG signal from each file from the database, since a separate file containing ECG signal is created for every patient recording.

Step 2: next we filter the files so we can only get the signal data without the text for the timestamp. The software used with the sensors records timestamp with every ECG signal recording.

Step 3: we resample the signal to enable economically using each data sample and to improve its accuracy, by using the Fourier method.

For most of the signals we follow the recommendation given by Heartpy's documentation to filter and resample the signal before processing it. For the rest of the files a Bad Signal Warning was generated. This warning pointed some possible corruptions that lead to lack of processing: detected heart rate falls outside of bpmmin - bpmmax constraints, no detectable heart rate present in the signal or very noisy signal.

Step 4: We process the signal and obtain/calculate heart and respiratory rate. We use Heartpy's functions to filter and resample the signal as well as to process it. By processing the signal using Heartpy an array of measurements is returned. The specific measurements that we are interested in are "bpm" (beats per minute) and "breathing rate". The breathing rate or also known as respiratory rate is measured in Hertz (Hz). These two values are also referred to as the *calculated values*.

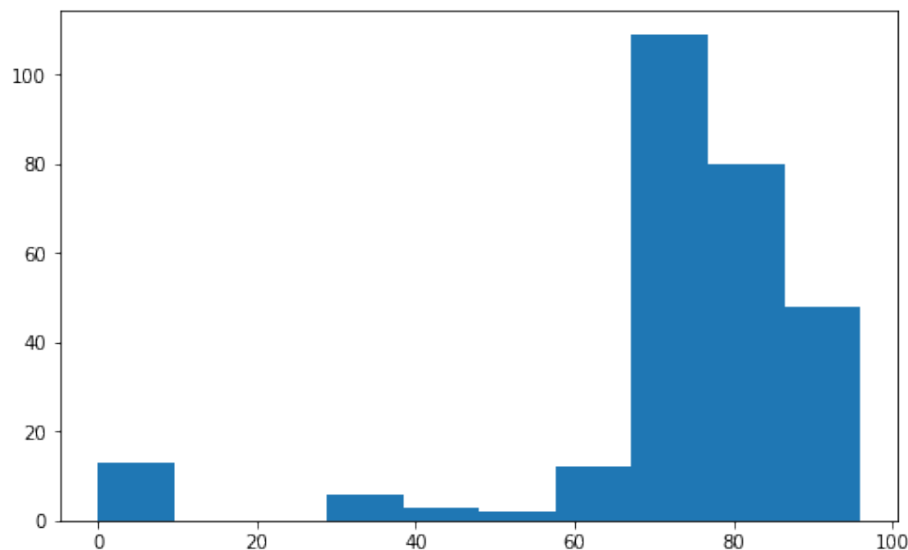


Fig. 3. Example of bad signal - histogram for heart rate.

Step 5: The calculated HR and RR values are compared with the measured values for HR and RR from the same patients, at the same time with the measured ECG signal. The measured values are the signals from the database itself which represent the real life measurements.

- To preprocess the data for comparison with the calculated values, first we read the recorded data for HR and RR that are contained in separate files. Since HR is measured for 30 seconds, but we need only one value for the patient, we find the mean value of the recorded measurements.
- As for the RR, since the biosensor measures it as breaths per minute, we multiply the RR outcome by 0.016667, because 1 hertz is equal to 60 Beats per minute. We need this conversion to express the outcome in Hz, because of the comparison with results obtained from the Python library.
- After this preprocessing, both the measured values and the calculated values are ready for comparison.

3.2 Visualizing the signals

Data visualization is an important aspect of data comprehension and analysis. It offers a way to communicate information that the human mind naturally comprehends. Due to the amount of data, visualisation is an important tool in this process. The data are visualized using the following libraries: Heartpy, Matplotlib and Neurokit2.

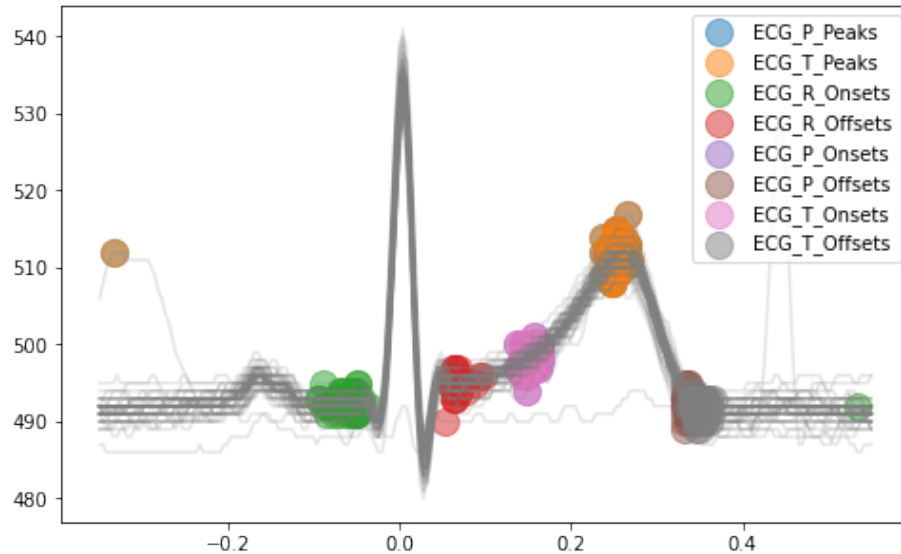


Fig. 4. An example visualization using the Neurokit2 library.

By using Heartpy's plotter we visualize the ECG signal. An example signal is given in Figure 1. We visualize each of the ECG signals in order to see if it is a corrupted or usable one and to check how many peaks are accepted by Heartpy in the calculations. The corrupted ECG signals are being filtered manually.

In order to make an even deeper analysis, we plot a histogram for both the HR and RR signals using Matplotlib. By plotting the histograms we can find out which HR and RR signals have missing values, outliers or even extreme values. Example of bad signal values for respiratory rate and heart rate are given in Figure 2 and Figure 3 respectively. These are considered as bad values because '0' as a signal value is unacceptable since this would mean that the subject's respiratory or heart rate is value 0. This value indicates that the subject is not breathing or the heart of the subject is not beating at all. Some of the signals that have missing values and outliers are accepted and some are filtered out.

Another analysis is conducted on the line plots for both the HR and RR signal. The line plots show if there are abrupt drops or increases in the signals that are unacceptable. Example drops are shown in Figure 5 and Figure 6.

The final visualization that we plot is a stacked signal for every QRS interval. This visualization is conducted using the library Neurokit2. The visualization shows how stable is the patient's ECG signal or in other words how repetitive is the patient's heart within the measuring time frame. An example visualization using the continuous wavelet method and the Neurokit2 library is given in Figure 4.

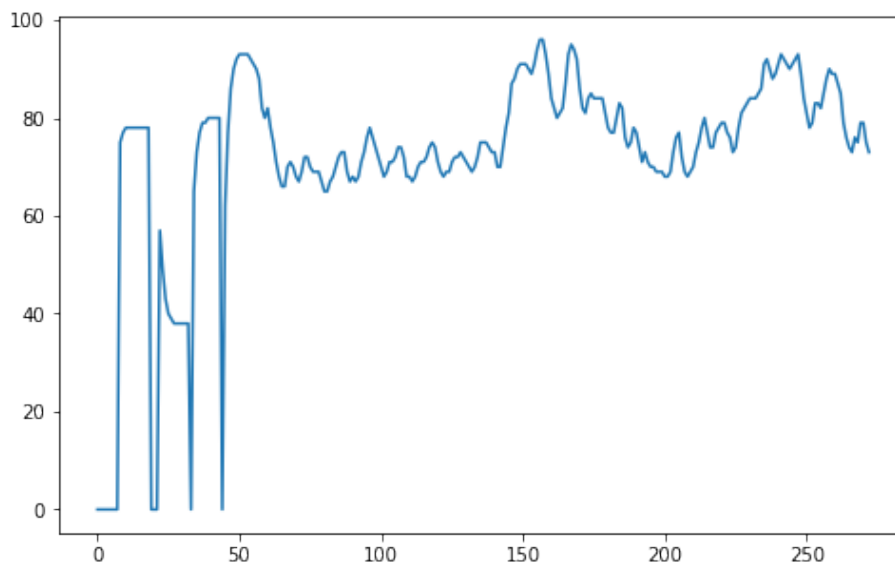


Fig. 5. An example drop in heart rate

4 Results

In order to validate the connection between the measured and calculated HR and RR, we conduct two types of analysis. The first one is calculating basic root mean squared error (RMSE) between the calculated and the measured values for both the HR and the RR. The RMSE value for the HR signals is 0.7 after all of the filterings and all of the processings. This shows that using the Heartpy library we can predict a very accurate heart rate signal just by using the ECG signal. On the other hand, the RMSE value for the RR is 0.113 Hz or 6.83 breaths per minute. This result is not satisfying. The typical value for the respiratory rate in humans is within the range of 10 to 20 breaths per minute (0.16 - 0.33 Hz). This means that by predicting the RR signal using the ECG signal we can hypothetically predict that a patient is breathing with 24 breaths per minute which would mean abnormal breathing when in reality he was breathing with 18 breaths per minute, the normal RR.

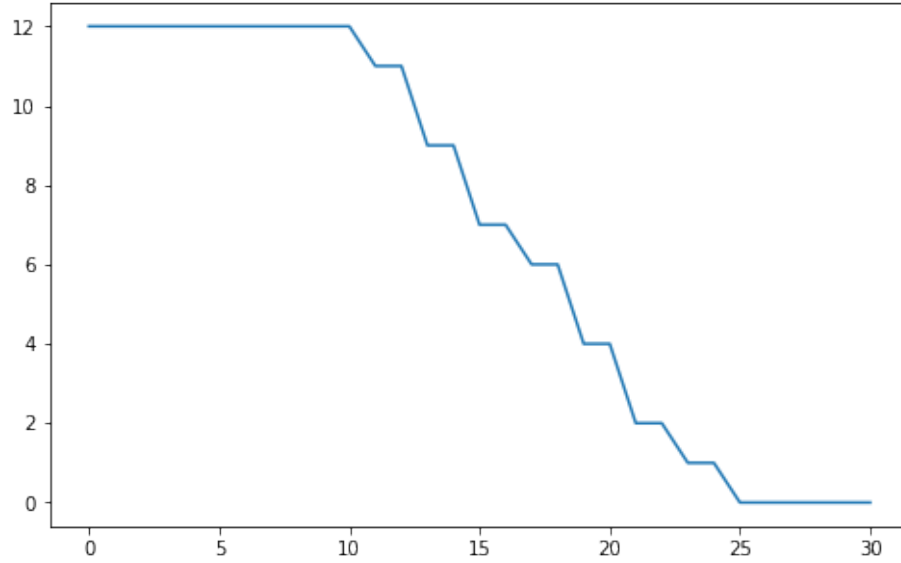


Fig. 6. An example drop in respiratory rate

The second conducted analysis is finding the correlation between the calculated and measured data. We achieve this by calculating the Pearson's and the Spearman's coefficient of correlation and by visualizing the scatter plot for the data. The Pearson's and the Spearman's coefficient for the HR are 0.998 and 0.997 respectively and the scatter plot is given in Figure 7. This shows that the measured and the calculated data are almost perfectly the same. On the other hand, the Pearson's and the Spearman's coefficient for the RR are -0.016 and -0.068 which means that the data is not correlated, as expected, taking into consideration the RMSE measure from the previous section. The scatter plot in Figure 8 shows the same result.

5 Discussion

The data quality is very important in the modeling process. Some actions were undertaken in the database, in order to improve the HR and RR correlation.

The obtained results were optimized by excluding some files from our database that weren't compatible with the rest, because using these files produced errors that corrupted the correlation coefficients. In order to filter the data files, they were categorized in 3 parts: *Old errors*, *Possible errors* and *Definite errors*.

The *Old errors* were found during the preprocessing of the files (only 2 files), which when filtered lost the information needed from the signal, leading to bad results. To avoid this problem, we singled out those files and continued without undertaking the filtering process on them.

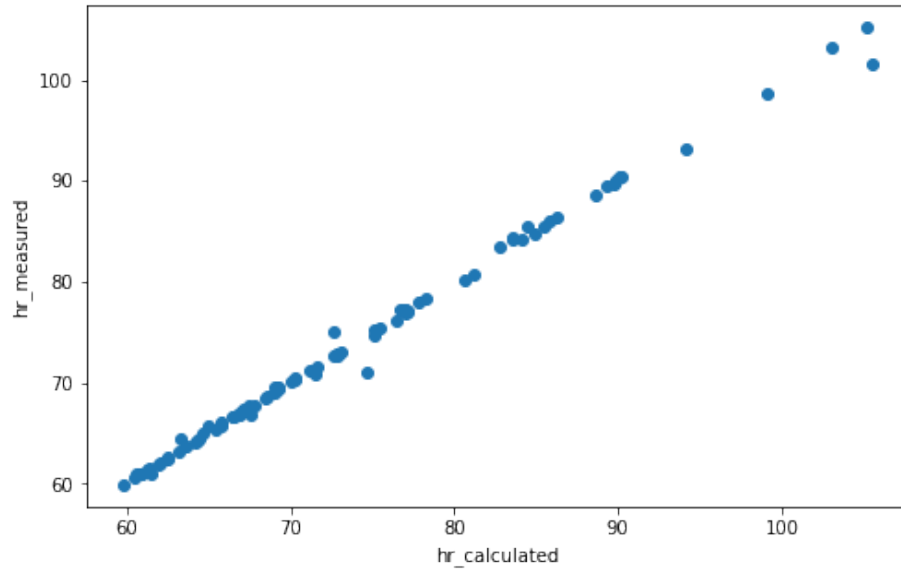


Fig. 7. Scatter plot for the HR

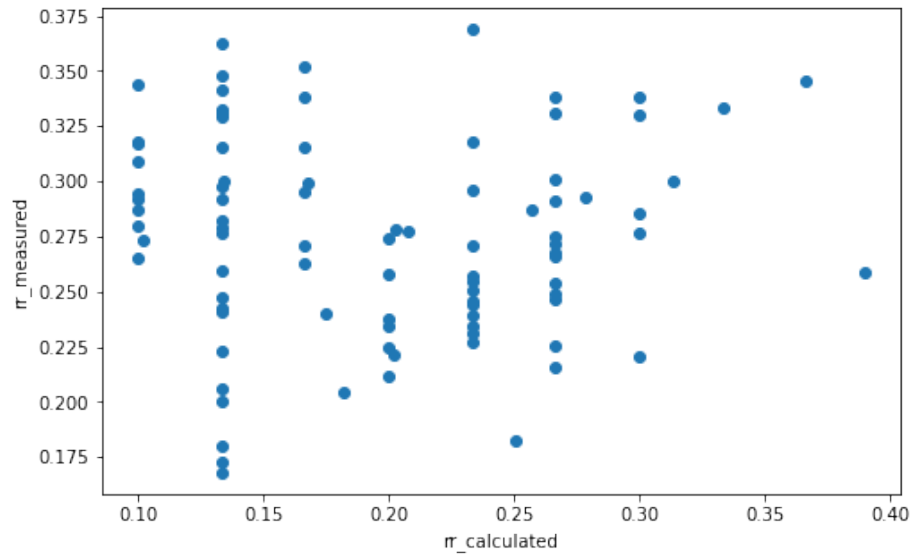


Fig. 8. Scatter plot for the RR

The files in the category *Possible errors* were found after the preprocessing and filtering. The measured heart rate compared to the calculated heart rate of the ECG signal had a high accuracy, however the calculated respiratory rate compared with the measured respiratory rate had some inconsistencies. Our opinion is that this might be due to some noise and artifacts not filtered in the signal, as well as the inconsistencies in the Q,R and S peaks of the ECG signal.

The files in the *Definite errors* were also found after the preprocessing and filtering, but as we can see from the images, the signals are ill measured by the sensor, which led to bad results. Some of them gave us a calculated respiratory rate equal to 0, and some gave us calculated respiratory rate way above the interval for normal resting respiratory rate. Those files were removed from the database.

However, all of this errors were captured on only few files from our database, while the rest gave us much accurate calculations. We can see from the results that the correlations' coefficients are positive, which means that we can rely on the ECG signal to calculate the Heart Rate in future research using the Python libraries. Further analysis is needed to determine the reason for the Respiratory Rate correlation results.

6 Conclusion

This paper explore the algorithms that can be used to calculate the Heart Rate only by using an ECG signal from a patient, using Python libraries. Furthermore, we showed the importance of processing the data and ensuring quality data set. We presented several ways of cleaning the data by excluding the few files from the database which were corrupted. The calculated correlations' coefficients (Pearson's and Spearman's) between the calculated and measured HR is above 0.9. These results are based on the good quality of the data that was collected, preprocessed and filtered before the measurements were calculated. The quality of the data is what makes this algorithm an appropriate one for this research. From this we can come to a conclusion that the ECG signal is reliable and the filtered data can be further used for evaluation of other algorithms as well.

For future work, we will analyze and explore bigger data sets to see if there are suitable coefficients to scale the RR outcome of the Zephyr sensor, to improve the correlation with the calculated RR using Python libraries.

Acknowledgment

This paper has been written thanks to the support of the Smart Patch for Life Support Systems (NATO project G5825 SP4LIFE)

References

1. Birrenkott, D.A., Pimentel, M.A., Watkinson, P.J., Clifton, D.A.: Robust estimation of respiratory rate via ecg- and ppg-derived respiratory quality

- indices. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 676–679 (2016). <https://doi.org/10.1109/EMBC.2016.7590792>
2. Bowers, E., Murray, A., Langley, P.: Respiratory rate derived from principal component analysis of single lead electrocardiogram. In: 2008 Computers in Cardiology. pp. 437–440 (2008). <https://doi.org/10.1109/CIC.2008.4749072>
 3. Chan, A.M., Ferdosi, N., Narasimhan, R.: Ambulatory respiratory rate detection using ecg and a triaxial accelerometer. In: 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 4058–4061 (2013). <https://doi.org/10.1109/EMBC.2013.6610436>
 4. van Gent, P., Farah, H., van Nes, N., van Arem, B.: Heartpy: A novel heart rate algorithm for the analysis of noisy signals. *Transportation research part F: traffic psychology and behaviour* **66**, 368–378 (2019)
 5. Gursky, E.A., B., H.: *Handbook for Pandemic and Masscasualty Planning and Response*, vol. 100. IOS Press (2012)
 6. Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M.H., Brett, M., Haldane, A., del Río, J.F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E.: Array programming with NumPy. *Nature* **585**(7825), 357–362 (Sep 2020). <https://doi.org/10.1038/s41586-020-2649-2>, <https://doi.org/10.1038/s41586-020-2649-2>
 7. Hunter, J.D.: Matplotlib: A 2d graphics environment. *Computing in Science & Engineering* **9**(3), 90–95 (2007). <https://doi.org/10.1109/MCSE.2007.55>
 8. Kim, J.M., Hong, J.H., Kim, N.J., Cha, E.J., Lee, T.S.: Two algorithms for detecting respiratory rate from ecg signal. In: Magjarevic, R., Nagel, J.H. (eds.) *World Congress on Medical Physics and Biomedical Engineering 2006*. pp. 4069–4071. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
 9. Nussinovitch, U., Elishkevitz, K.P., Katz, K., Nussinovitch, M., Segev, S., Volovitz, B., Nussinovitch, N.: Reliability of ultra-short ecg indices for heart rate variability. *Annals of Noninvasive Electrocardiology* **16**(2), 117–122 (2011)
 10. Orphanidou, C., Fleming, S., Shah, S., Tarassenko, L.: Data fusion for estimating respiratory rate from a single-lead ecg. *Biomedical Signal Processing and Control* **8**(1), 98–105 (2013). <https://doi.org/https://doi.org/10.1016/j.bspc.2012.06.001>, <https://www.sciencedirect.com/science/article/pii/S1746809412000699>
 11. P. van Gent: Python heart rate analysis toolkit documentation release 1.2.5 - 20.07.2021 (2021), <https://readthedocs.org/projects/python-heart-rate-analysis-toolkit/downloads/pdf/latest/>
 12. Parák, J., Havlík, J.: Ecg signal processing and heart rate frequency detection methods. *Proceedings of Technical Computing Prague* **8**, 2011 (2011)
 13. Ruangsuwana, R., Velikic, G., Bocko, M.: Methods to extract respiration information from ecg signals. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 570–573 (2010). <https://doi.org/10.1109/ICASSP.2010.5495584>
 14. Sinnecker, D., Dommasch, M., Barthel, P., Müller, A., Dirschinger, R.J., Hapfelmeier, A., Huster, K.M., Laugwitz, K.L., Malik, M., Schmidt, G.: Assessment of mean respiratory rate from ecg recordings for risk stratification after myocardial infarction. *Journal of Electrocardiology* **47**(5), 700–704 (2014). <https://doi.org/https://doi.org/10.1016/j.jelectrocard.2014.04.021>, <https://www.sciencedirect.com/science/article/pii/S0022073614001538>

15. Zephyr Technology: Zephyr bioharness 3.0 user manual (2017), <https://www.zephyranywhere.com/media/download/bioharness3-user-manual.pdf>