Deep Learning

ECG Classifier - WESAD - Data Pre-Processing

CASTETS Edouard - A20496836

Date : September 9, 2022

1 Construction of the ECG dataset

1.1 Feature Extraction

The WESAD is a dataset built by Schmidt P et al[2] because there was no dataset for stress detection with physiological at this time.
Among the measures, the dataset contains Electrocardiogram measures of 15 subjects during 2hours with stressing, amusing, relaxing and neutral situation. The ECG is measured with a frequency of 700Hz. This is a 20s sample from the dataset:

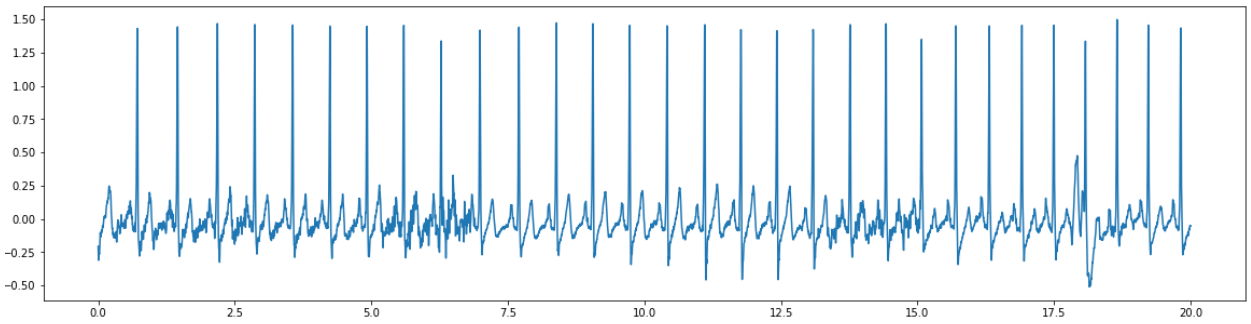

Figure 1: ECG sample of 20s

As they described in the WESAD paper, there are relevant features to extract from this signal to detect stressing situation. To extract these features, I choose to segment the all signal (2 hours) to sample of 20s, thus for inference the stress detection based on ECG could probably be based on the detection of stress in a measure of 20s on the student. It was essential for me to detect peak from the signal. I firstly tried to implement this function by myself with a classical algorithm of peak detection, yet some samples from the dataset have huge variation in the peak amplitude (probably due to a sensor malfunctioning). Thus I decided to use a python package named HeartPy, which aims to detect heart rate with peak position by using different filters before the algorithm of detection. I decided to use HeartPy for the rest of the features extraction to avoid wrong detection in some samples.

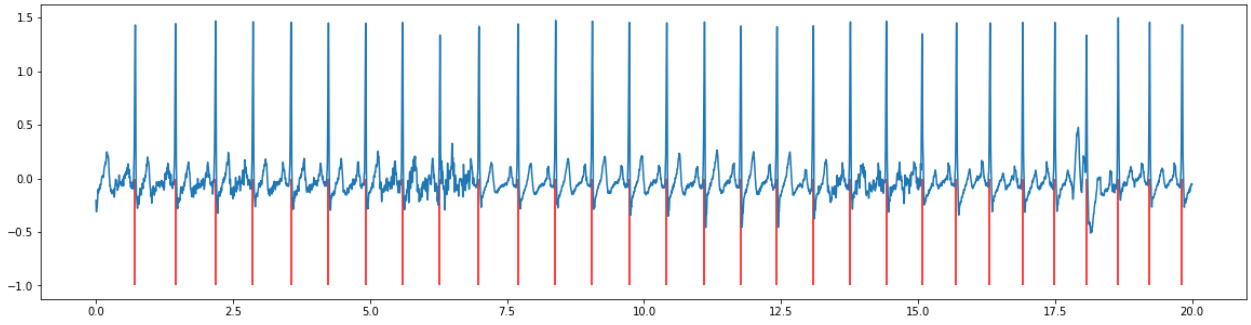

Figure 2: ECG peak detection via my own function

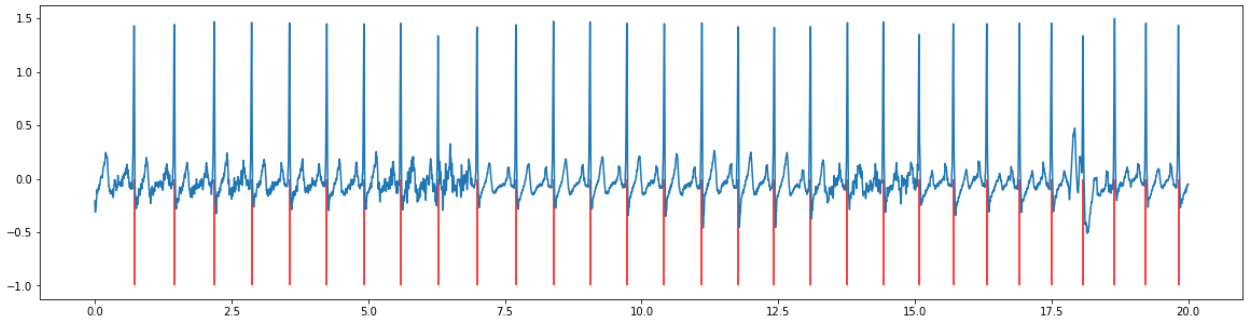

Figure 3: ECG peak detection via my heartpy function

From peak detection I was able to calculate statistical features from the heart rate and to calculate heart rate variability which is calculated as the time to go to the peak i to i+1. Then I was able to extract some statistical features.

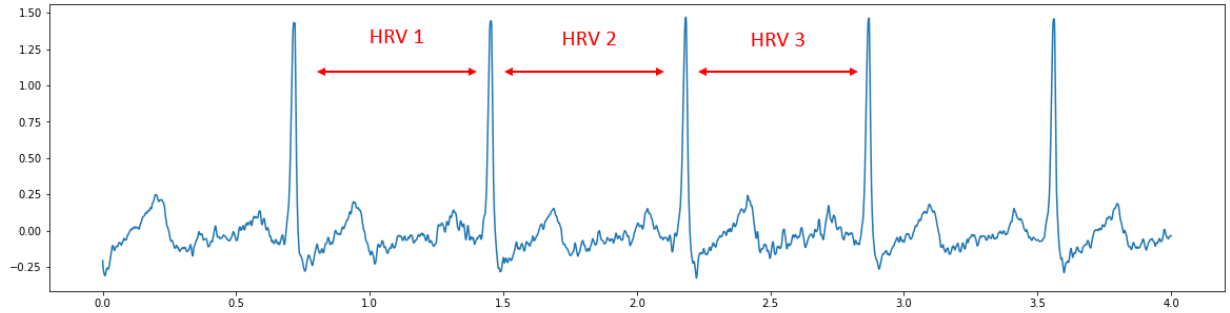

Figure 4: HRV calculation

I fitted a gaussian distribution to the HRV data using scipy in order to apply a triangular interpolation of this distribution, as it is mentioned in this paper from Malik et al[1]. This geometrical interpolation would reveal features relative to stress detection as TINN and HRV index that I computed as mentioned in their paper :

$$Y = max(distribution) \quad ; \quad HRV_{index} = \frac{totalnumberofpeakintervals}{Y}$$

The TINN score is computed from the calculation of M and N which are such as a multi-linear function of time q(t) is defined as q(t)=0 for t<N and t>M and $q(argmax(distribution)) = Y$ such as :

$$\int_0^\infty (D(t) - q(t))^2 dt$$

is minimum. Then $TINN = M - N$

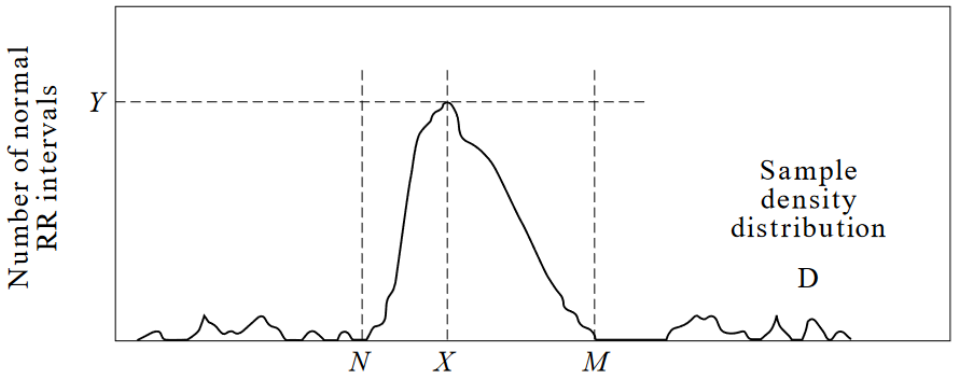

Figure 5: Distribution of HRV with TINN

I implemented the interpolation in Python and calculated HRVindex and TINN.I have also calculated the number of interval differing from more than 50ms (number and percentage).

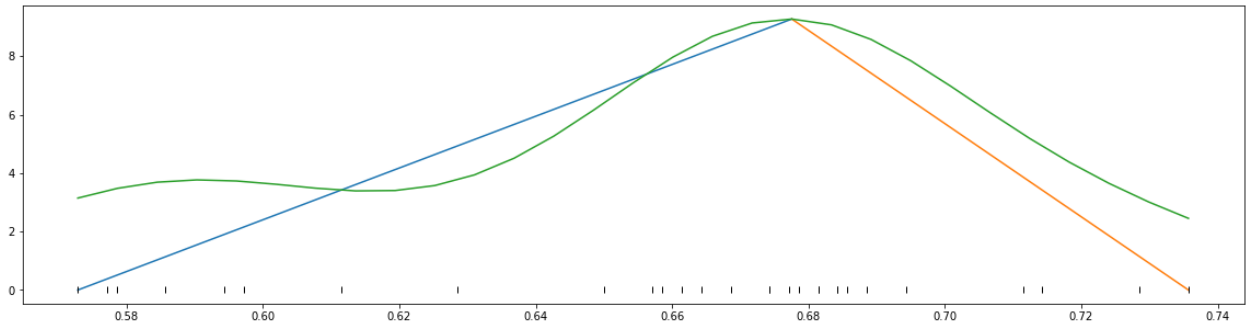

Figure 6: Triangular Interpolation on HRV

Finally I used the FFT, with a window size based on the mean of HRV, to extract the Power spectral density and I took the mean and standard deviation of Fourier frequencies (which is a mistake because it is not really depending of the sample except from the window size). I calculated the sum of the PSD components as a feature.
Finally I extracted these features :

Features	Units
Mean Heart Rate	Beat/s
STD Heart Rate	Beat/s
TINN	s
HRVindex	None
#NN50	None
pNN50	None
Mean HRV	s
STD HRV	s
RMS HRV	s
Mean Fourier Frequencies	Hz
STD Fourier Frequencies	Hz
Sum PSD components	None

These features will constitute my features vector for detection.

1.2 Dataset creation

As I said I extracted samples of 20s with a 1s step from every recording, these samples were coupled with a label : 1=neutral ; 2=stress ; 3=amusement ; 4=meditation. As ECG is very person dependant, I selected a 90s of the baseline, extracted the features and for every 20s sample I divided the features of the sample by the features of the baseline to have a comparison of the sample with a neutral moment from the baseline.

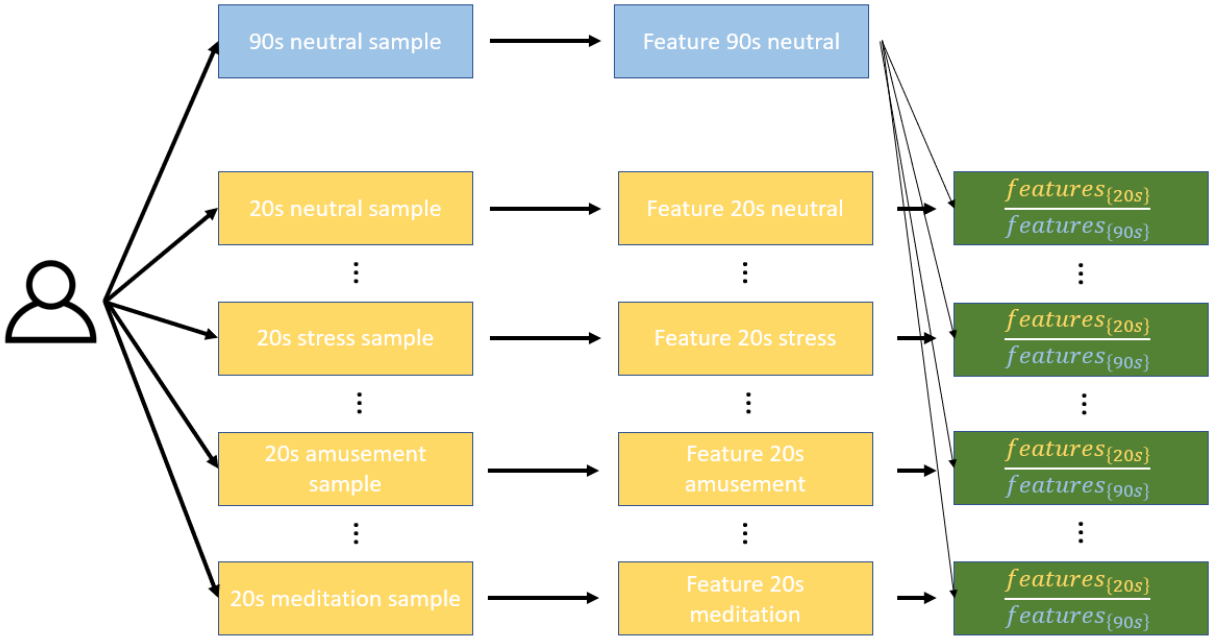

Figure 7: Features extraction for dataset

At the beginning I randomly choose samples for training, testing and validation among all the

samples from all the people, yet I figured out that a better treatment would avoid over-fitting. I firstly choose to split subjects between training, validation and testing because taking 2 different samples from a same subject for training and validation could create overfitting because my time step was 1s on 20s windows (for data augmentation).

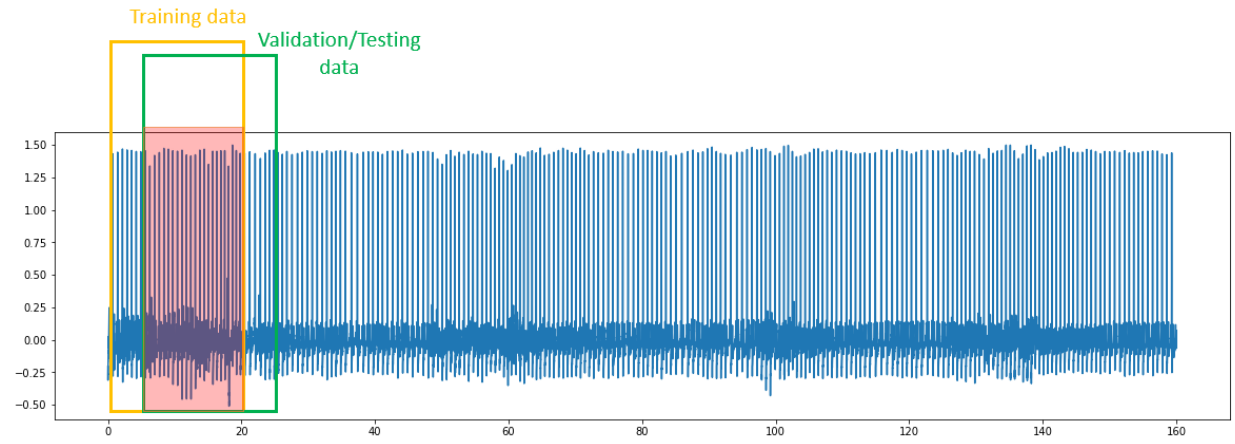

Figure 8: Possible Over-fitting

Thus I split the data of the 15 subjects into training, validation and testing. Subjects for training and validation will be permuted as I planned to use K-fold cross validation (2 subjects in validation, 12 in training), so 91 possible dataset. I choose subject 17 to be my testing subject and I never included this subject in the creation of the fold datasets.

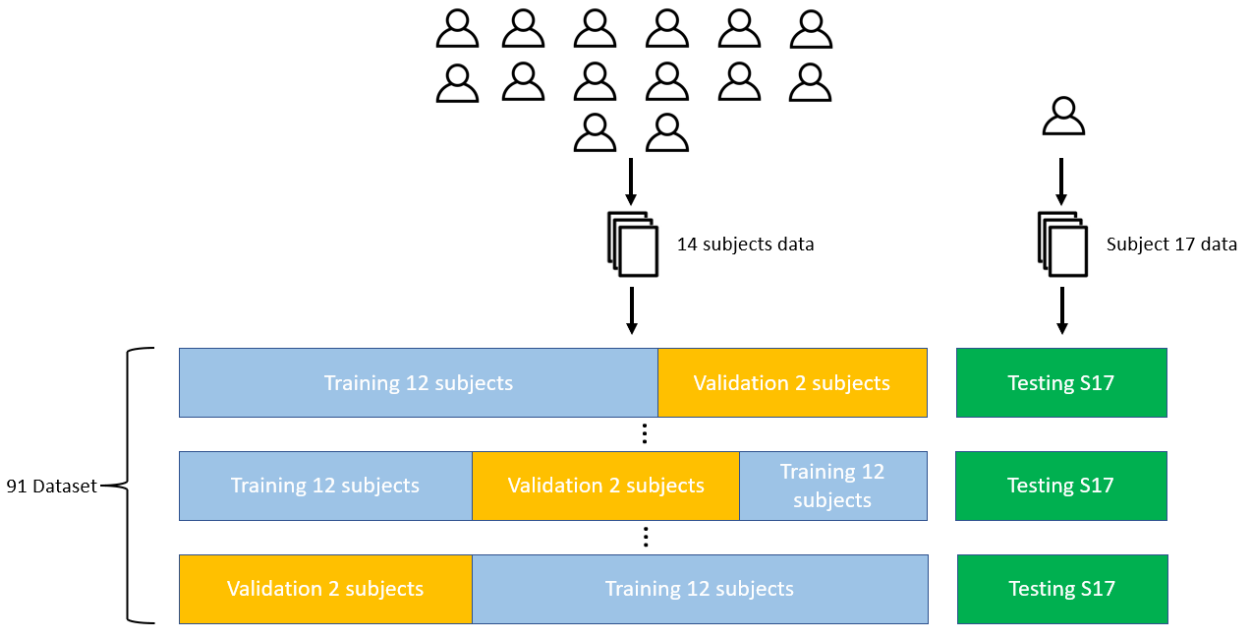

Figure 9: K-fold Dataset

Finally for each created Training dataset, I have chosen to discard weird data (HRV of 2s for example) due to malfunctioning of sensor creating troubles in the peak detection. I also have chosen to balance the data set to have 50% of stress data and 50% of non-stress data, to improve learning.

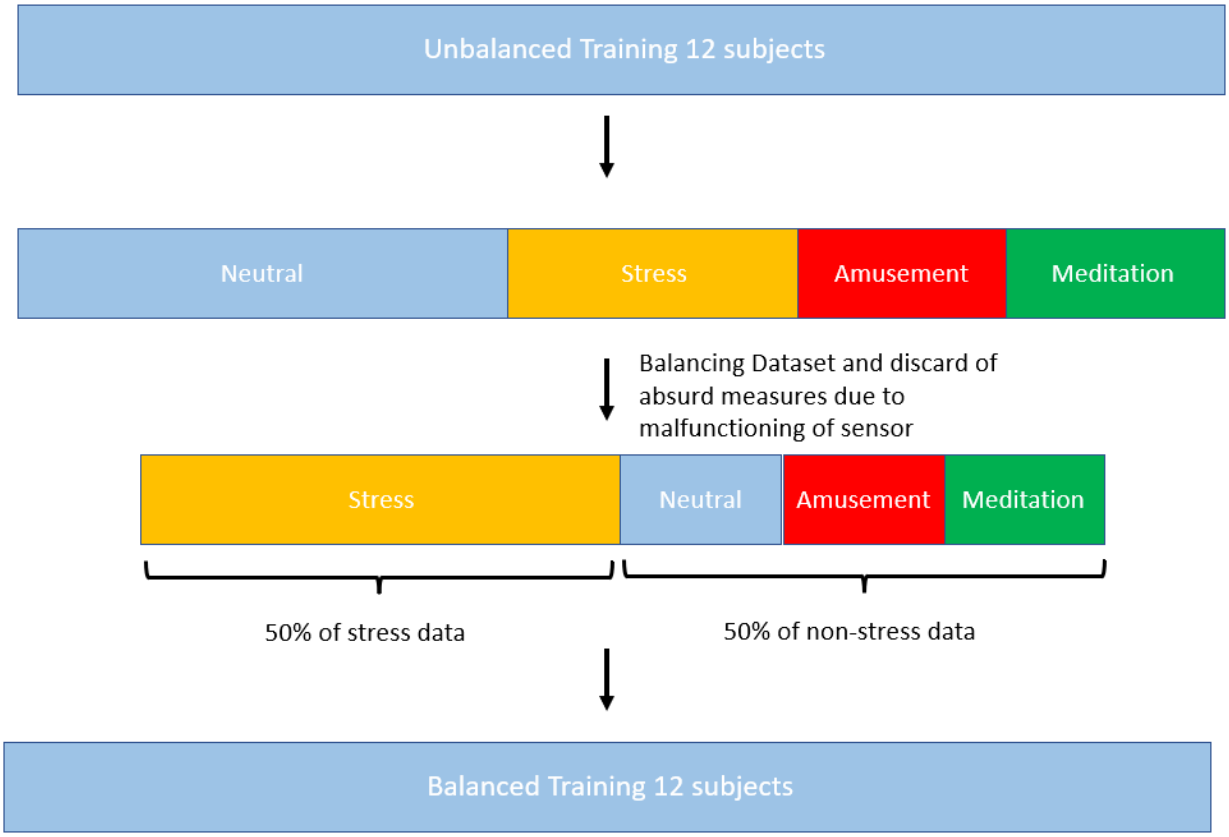

Figure 10: Balancing Dataset

.

References

[1] Marek Malik et al. "Heart Rate Variability". In: *Circulation* 93 (5 Mar. 1996), pp. 1043–1065. ISSN: 00097322. DOI: 10.1161/01.CIR.93.5.1043. URL: https://www.ahajournals.org/doi/abs/10.1161/01.CIR.93.5.1043.

[2] Philip Schmidt et al. "Introducing WeSAD, a multimodal dataset for wearable stress and affect detection". In: *ICMI 2018 - Proceedings of the 2018 International Conference on Multimodal Interaction* (Oct. 2018), pp. 400–408. DOI: 10.1145/3242969.3242985.