Internship write up

damian french

July 2025

1 Building Basic Inputs

The Majority of the Heart Rate data from the smart watch was stored in a .csv file called **raw_hr_hr.csv**, which contained 3 columns: Start, Duration, and Value as shown in the example below:

Table 1: Raw HR csv example

Start	Duration	Value
2025-07-17T20:23:09+02:00	"[3,3]"	"[126,126]"
2025-07-17T20:23:01+02:00	"[2,2,2]"	"[123,122,124]"
2025-07-17T20:22:39+02:00	[3]	[124]
2025-07-17T20:22:26+02:00	"[2,2]"	"[130,129]"

The column *Duration* was ignored and the columns *Start* and *Value* were filtered to be useable. For *Start* this was done by converting the column *Start* to a pandas DateTime series, and then sorting the HR data by date to order the data chronologically. For *Value*, each row in the column was split on commas, converted to a string, and then stripped of any brackets. All values were then convered to integers and averaged across each row, resulting in a single HR value for each timestamp.

There was also data available on the activities recorded by the watches in a .csv file called **activities.csv**, and this was parsed in a similar way along with the sleep data found in the file **sleep.csv**

Once this was done, the data was split up into months, weeks, active days, days and nights. This required formatting of the timestamps to fit the levels of time precision. The ability to plot these was added with Flags that can be updated to indicate whether or not the plots are desired.

The heart rate at night was determined using averages of sleep and awakening times. The **sleep.csv** file contained the times the patient went to sleep and woke up each day, and from these averages were taken using a circular mean. The times were mapped from 24 hours into 2π radians as unit vectors and then weighted with the times spent sleeping. The vectors were then

averaged and the resultant vector converted back into hours to determine the average sleeping and awakening times.

All this data was then returned as a dictionary, where each feild contained multiple arrays.

2 Detrended Fluctuation Analysis

Detrended fluctuation analysis is a modified RMS analysis of random walks used to determining long range correlations in nonlinear data, as set out by Peng et al [1]. The results of the DFA analysis are plotted on a log-log graph, and it is expected that there is a change in gradient at a certain point, when the long range correlations in the data start to overpower the local fluctuations. An example of a DFA plot generated from the patient data is shown in Fig.1. The Human heartbeat has a certain amount of fluctuation intrinsic to it, and in DFA this produces a long range gradient of around 1, indicating a power law relationship called pink noise. A gradient of 0.5 indicates white noise - complete randomness, and 1.5 indicates brown noise.

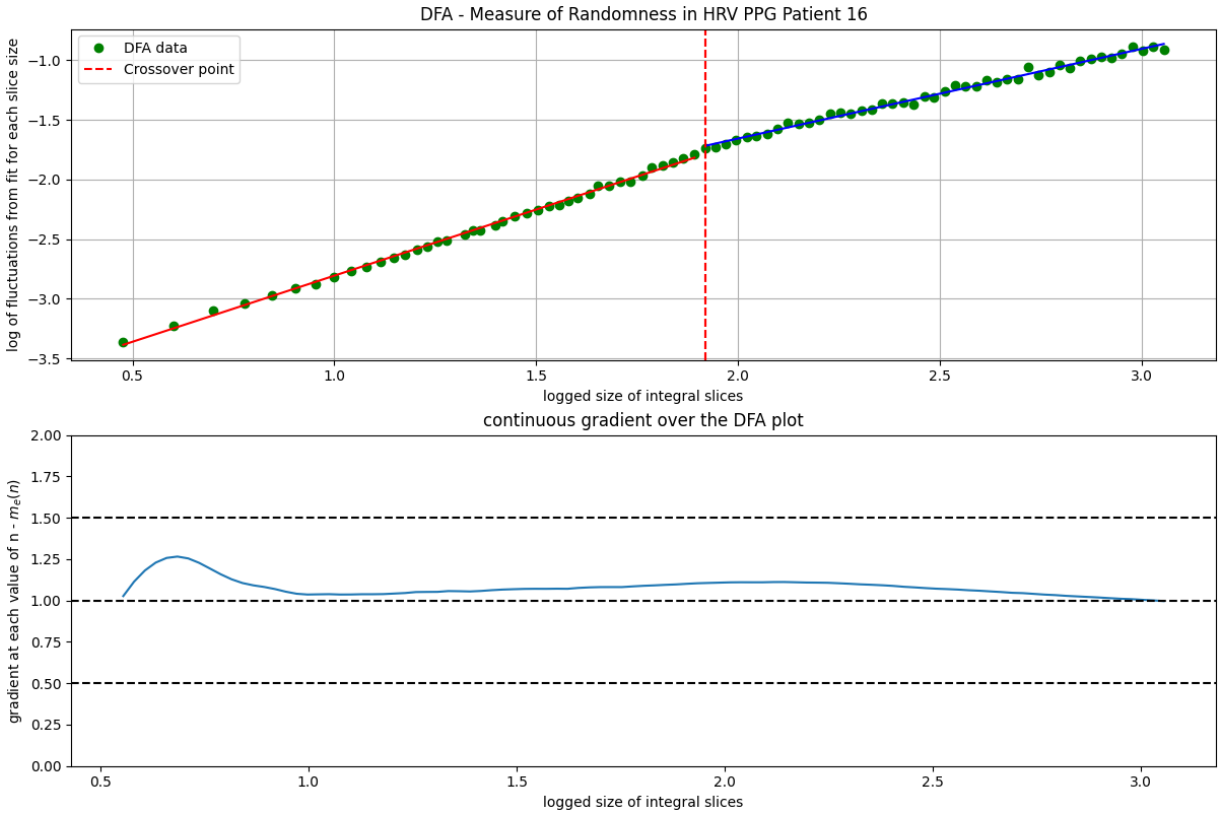

Figure 1: DFA plot for a patient who has undergone radiotherapy for lung cancer. The red dashed line indicates the crossover point. The top plot is using the raw data, whilst the bottom plot shows the gradient over the graph

The crossover phenomenon, which is the occurrence of the change in gradient, is explained qual-

itatively by Peng et at [1], and further quantitively by Hu et al [2].

DFA is only ever usually undertaken using ECGs, usually 10 or 12 lead. The attempt here was to see if the analysis could be done using

1. a single lead ECG
2. a smartwatch PPG

DFA works with Heart Rate Variability, i.e. the intervals between beats, which is why it is usually done with ECGs, since the RR peaks intervals of an ECG is a very good metric for HRV. In order to produce HRV data from the PPG, we inverted the *Values*, essentially turning them from beats per minute to minutes per beat. This gave us a much more imprecise HRV than an ECG, since each value is an average of beat intervals over a couple minutes, compared to the individual beat precision from an ECG. However, since the patients wore the watches for around 2 months, The number of data points is much greater than for the ECGs, and so this compensates for the lack or precision in the data.

Hu et al [2] show that an underlying linear trend in a signal will cause the DFA graph to change gradient at a certain *crossover scale*, therefore it is expected that for a heart, and one that may be deteriorating, would produce a DFA graph with a decrease in gradient, referred to as the *scaling exponent*, after the *crossover scale*. This can be ovserved in Fig.1 and when implemented for all patients using this method of HRV from PPG data, it can be seen in Fig.2 that there is a general trend towards a decrease in gradient across all patients.

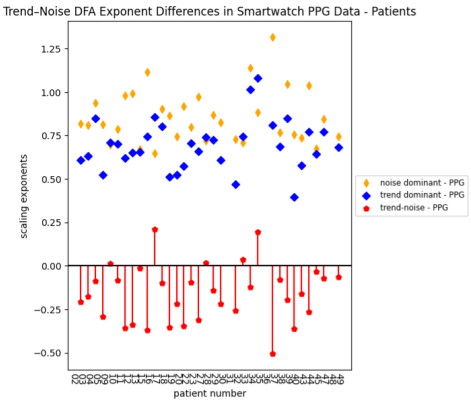

Figure 2: Plot of difference in gradients (*scaling exponents*) before and after the *crossover scale* for each patient using the PPG data. The yellow data points are the gradients before the crossover and the blue data points are the gradients after it. The red data points are then the difference between the two.

The mean difference in scaling exponent is less than 0, implying there is a decrease in gradient after the crossover scale throughout the population.

The bottom plot in Fig.1 is what is called a *scaling pattern*, as set out by Echeverria et al [3]. The scaling pattern gives more information about the *scaling exponent* throughout the graph, and can be used for further analysis of the HRV. It is expected that a healthy heart should asymptote towards a value of 1 on the graph, with few fluctuations after the beginning section, as explained Echeverria and Peng. An unhealthy heart should exhibit more variations in the later sections of the plot, and assymptote to a value either above or bellow 1, indicating either more or less randomness depending on the condition.

3 ECG implementation

The ECG data from the watches was stored in a .csv file **signal.csv**, and was analysed using previously written code to determine the R peaks and therefore the RR intervals. An attempt was made to try and clean the ECG signals, as the raw data is very noisy and can lead to missed or false R peaks.

Fourier analysis was used to attempt to remove higher frequencies from the signals, however this produced gibbs phenomenon which ended up increasing the noise rather than decreasing it. Wavelet thresholding was also attempted, which is meant to remove short range fluctuations and keep general trends, using auto correlation to find the dominant period and this was used to create an adaptive threshold for filtering. The smallest coefficients were removed from the highest frequency wavelets. There was more success with this method than the fourier one, with more detail on this needed, but an example of the results is shown in fig.3.

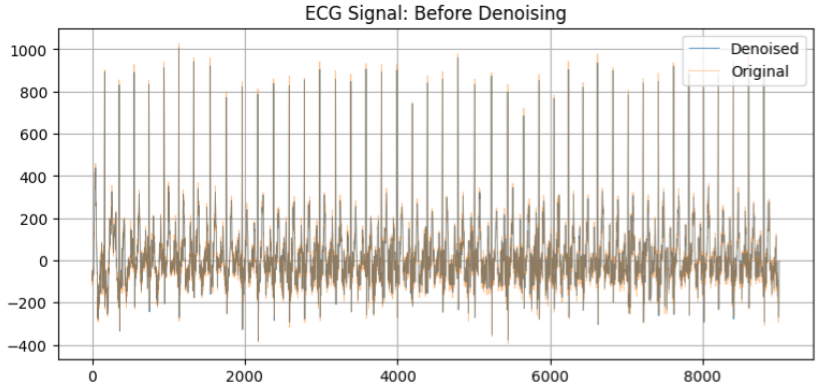

Figure 3: Caption

The ECG RR intervals were brought into the main pipeline and were combined into 1 long ECG for each patient. This was done to provide enough data for DFA, since each ECG was only

30 seconds long, resulting in only about 30 R peaks per ECG, and therefore 29 RR intervals. These new chained ECGs were then filtered for outliers, using a rolling mean of the previous 5 RR intervals and removing points more than 120% of that mean. This is a standard filtering technique used in many works and set out in the Advanced Methods and Tools for ECG Data Analysis Textbook [4]. Once this was done, DFA was then implemented on the ECGs. The results of this were consistent with the PPG results across the population, with a similar change in *scaling exponent* observed, as seen in Fig.4. A paired t-test on the PPG data with the null hypothesis of having the same mean as the ECG data gave a p value of 0.4, suggesting that we cannot reject the null, implying there is similarity between the distributions.

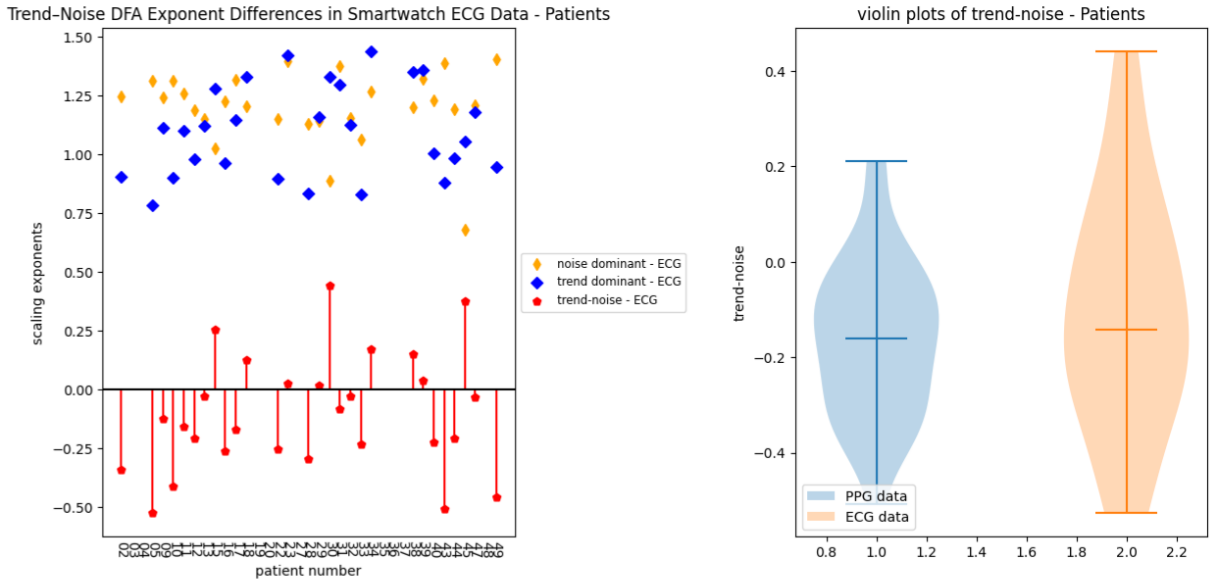

Figure 4: Plot (*left*) of difference in gradients (*scaling exponents*) before and after the *crossover scale* for each patient using ECG data. Plot (*right*) of violin plots for ECG and PPG difference in *scaling exponents*. The two distributions are similar with a very similar mean.

Using the *scaling patterns* an average across the ECGs and PPGs was found in order to compare the method for the two types of data. The results, as show in Fig.5, indicate that there is a similarity between the data, especially after the very short range correlations, which again, as indicated by Echeverria and Peng, is where the fluctuations are more varied since noise is more powerful in this region. There is arguably just a systematic shift between the two average *scaling patterns*, and this is further indicated by the reasonably linear line in the correlation plot, again ignoring the values taken from the beginning of the *scaling patterns*. The spearmans rank correlation coefficient between the two Averages was 0.3, again suggesting a correlation between the two.

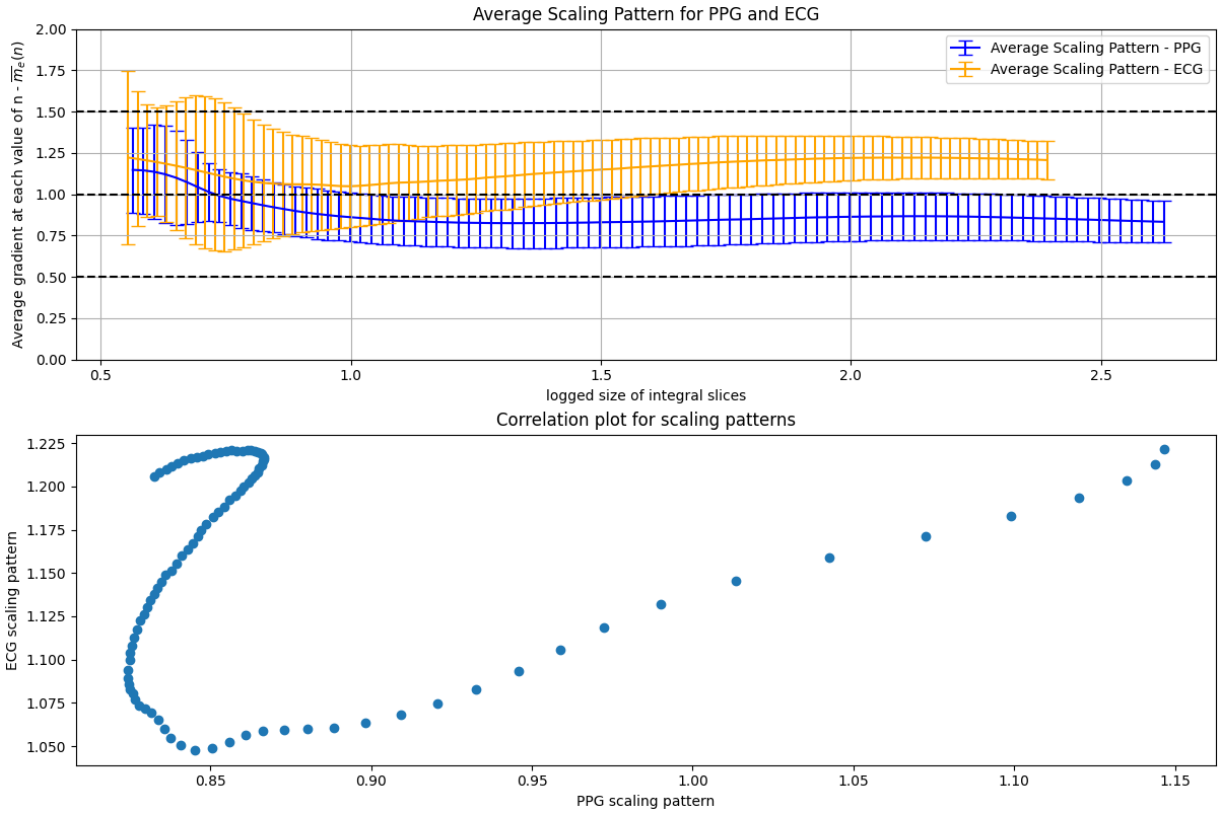

Figure 5: top Plot contains the averages of the PPG and ECG scaling patterns averaged over each patient. The bottom plot is the correlation between the two average scaling patterns.

As well as an average, *scaling pattern* difference between the ECG and PPG was also plotted as shown in Fig.6. This first plot agrees with Fig.5, where at the beginning there is much variance between methods, but as we move from shorter to longer range fluctuations, by increasing window size, the difference in methods trends towards 0, seeming to asymptote somewhere between 0 and 0.5. This is further observed with the second plot in fig.6, where the range in scaling pattern difference between ECG and PPG seems to be decreasing, although the axis is not long enough to tell for certain.

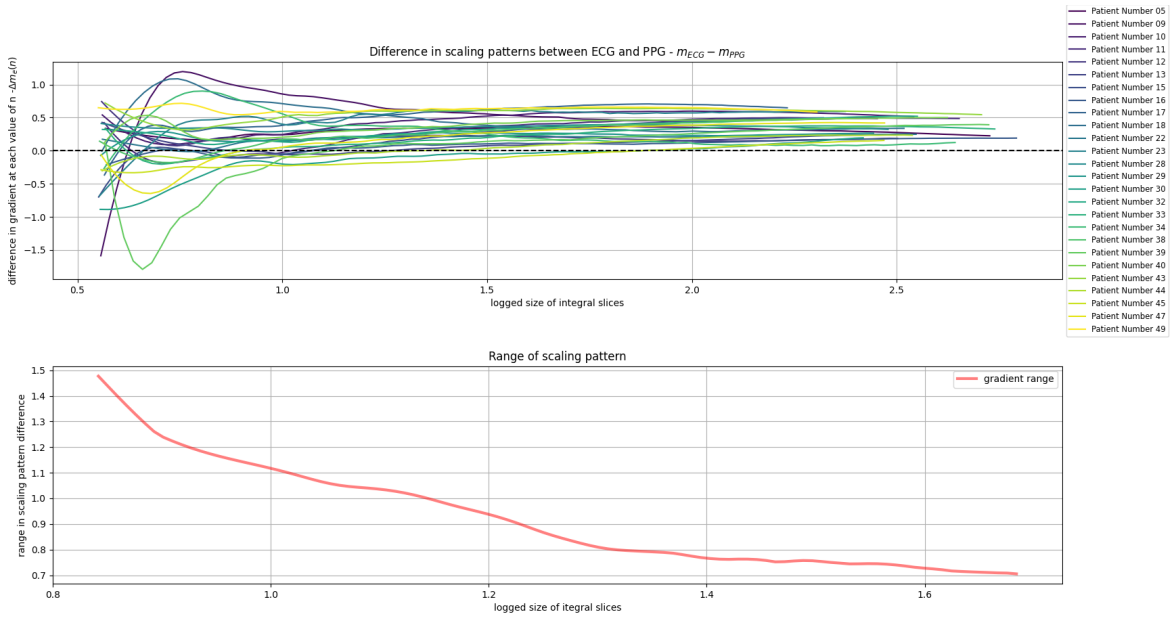

Figure 6: difference in scaling pattern between ECG and PPG for each patient in top graph. Bottom graph shows the difference between the best and worst agreeing scaling patterns in a certain region of the graph.

Fig7 again gives evidence of the correlation between ECG and PPG DFA.

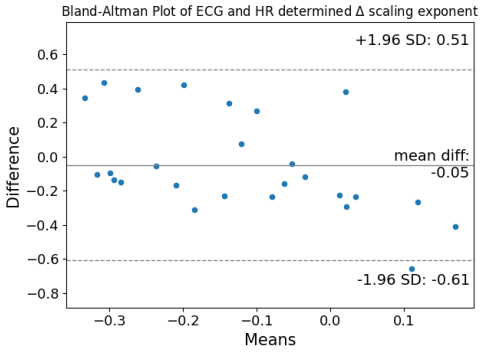

Figure 7: Bland Altman plotof difference in scaling exponent between ECG and PPG.

4 Database creation

All the metrics calculated were added to a dictionary with a row for each patient, and then this in turn was used to add to a database. An SQL database was created, and in such a way that it meets 3rd normal form relational database standard. The database diagram can be seen in Fig.8. This layout enables connections between many different metrics in a fast and logical way, and allows quick and easy analysis.

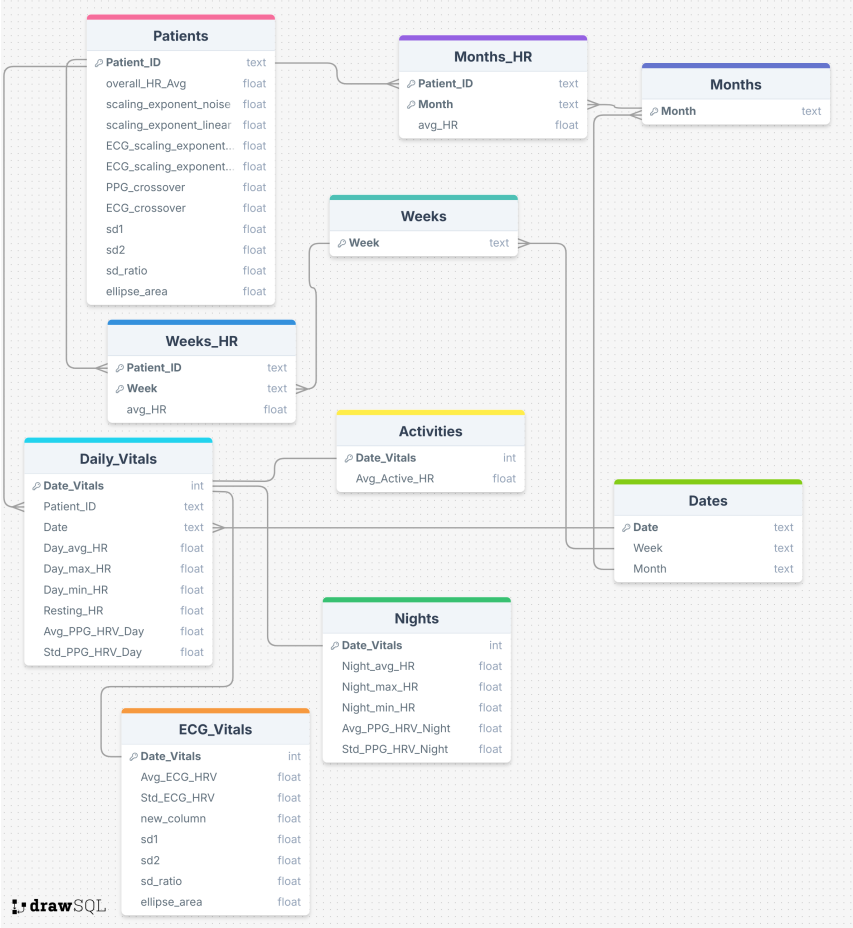

Figure 8: relational database used to store all the smartwatch heart metrics

5 Metrics Analysis

Using the database, many metrics can be plotted, such as weekly HR for each patient, and for the menial amount of volunteer data we collected, shown in Fig.9.
A heat map of the same data is shown in Fig.10, giving a better understanding on how the data changes over time across the population.

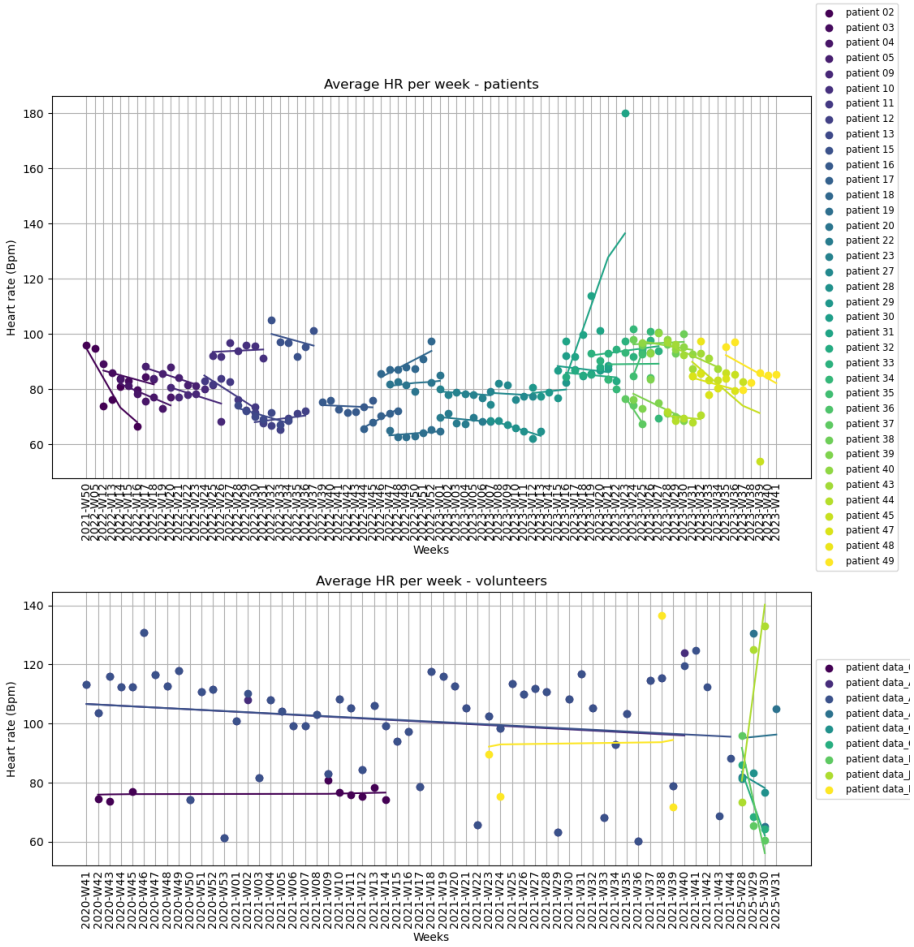

Figure 9: Weekly Heart rate for each patient on the top, for volunteers on the bottom

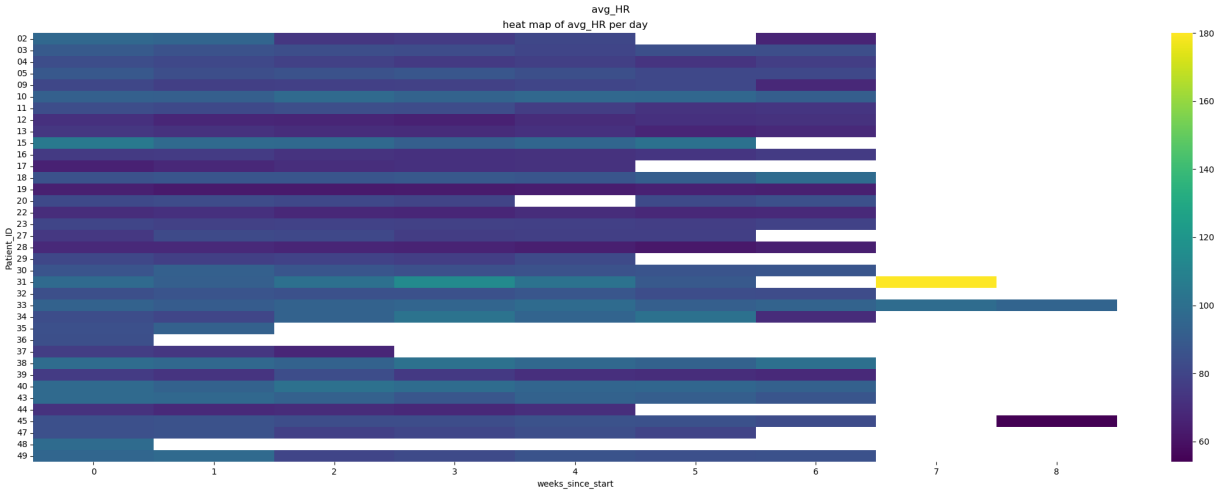

Figure 10: Heat map of average weekly heart rate for each patient

Much more complicated plots can be created, for instance connecting Weeks to activities to see how the active heart rate varies by week, or Night minimum Heart rate on days with activity and so on.

References

[1] C. Peng, S. Havlin, H. E. Stanley, and A. L. Goldberger, "Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 5, pp. 82–87, 03 1995.

[2] K. Hu, P. C. Ivanov, Z. Chen, P. Carpena, and H. Eugene Stanley, "Effect of trends on detrended fluctuation analysis," *Phys. Rev. E*, vol. 64, p. 011114, Jun 2001.

[3] J. C. Echeverrıa, M. S. Woolfson, J. A. Crowe, B. R. Hayes-Gill, G. D. H. Croaker, and H. Vyas, "Interpretation of heart rate variability via detrended fluctuation analysis and filter," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 13, pp. 467–475, 06 2003.

[4] G. C. Clifford, F. Azuaje, and P. McSharry, *Advanced Methods and Tools for ECG Data Analysis*. Norwood, MA, UNITED STATES: Artech House, 2006.