HRV: a Pythonic package for Heart Rate Variability Analysis

Rhenan Bartels[1] and Tiago Peçanha[2]

1 Pulmonary Engineering Laboratory, Biomedical Engineering Program, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil **2** Faculty of Medicine, University of São Paulo, Brazil

Summary

Heart rate variability (HRV) is a non-invasive tool to assess the cardiac autonomic integrity and cardiovascular homeostasis (Electrophysiology, 1996). HRV quantifies the instantaneous variations in the RR intervals, which is produced by the balanced action of the parasympathetic and sympathetic branches of the autonomic nervous system (ANS) over the sinoatrial (SA) node, and modulated by different physiological inputs (e.g., respiration, blood pressure, temperature, emotions, etc) (Malik & Camm, 1990). Specifically, the parasympathetic activation produces fast and short-lasting bradycardia, which results in high-frequency oscillations (i.e., 0.15 - 0.40 Hz) in heart rate. On the other hand, sympathetic activation produces slower and longer-lasting variations in heart rate, which results in low-frequency (i.e., $\sim$ 0.1 Hz) oscillations in heart rate.

Increased HRV indicates a predominance of the parasympathetic over the sympathetic activation in the SA node, and indicates enhanced cardiac autonomic flexibility and improved overall health (Malik & Camm, 1990). Conversely, a reduced HRV is usually accompanied by sympathetic dominance and suggests increased rigidity and loss of control of the ANS to the cardiovascular system, which is a sign of disease vulnerability (Malik & Camm, 1990). For instance, a reduced SDNN (i.e., standard deviation of RR intervals, a simple statistical-based time domain index of HRV) has been shown to overperform traditional cardiovascular risk parameters in predicting mortality in a cohort of heart failure patients (Nolan et al., 1998). A reduced HRV has also been linked with metabolic dysfunction (Weissman, Lowenstein, Peleg, Thaler, & Zimmer, 2006), increased inflammation (Sajadieh et al., 2004), depression (Sgoifo, Carnevali, Pico Alfonso, & Amore, 2015), psychiatric disorders (DeGiorgio et al., 2010), sleep disturbance (Burton, Rahman, Kadota, Lloyd, & Vollmer-Conna, 2010), among others. Based on this wide prognostic utility, the interest in approaches to evaluate HRV has shown an exponential growth in different medicine specialties and research fields in the recent years.

HRV is routinely assessed using linear methods, through the calculation of different indices either in time- or frequency-domain. Time-domain consists of a collection of statistical metrics, such as the average value of RRi (mRRi), the standard deviation of RRi (SDNN; the NN stands for natural or sinusal intervals), the standard deviation of the successive differences (SDSD), the number or percentage of RRi longer than 50ms (NN50 and pNN50) and the root mean squared of successive difference in adjacent RRi (RMSSD - equation 1) (Electrophysiology, 1996). Each of these indices quantifies different facets of the HRV, which are promoted by different autonomic sources. SDNN quantifies overall variability behind HRV, which is produced by both parasympathetic and sympathetic branches. NN50, pNN50, and RMSSD quantify beat-to-beat HRV, which is produced predominantly by the parasympathetic action in the heart.

$$RMSSD = \sqrt{\frac{1}{N-1}\sum_{j-1}^{N-1}(RRi_j - RRi_{j+1})^2}$$

- Equation 1

where N is the count of RRi values and RRi_j is the jth RRi value.

The frequency-domain analysis quantifies the extent of contribution of each frequency component to the overall heart rate fluctuation (Figure 2). The main frequency components are the VLF (i.e., very low frequency; < 0.04 Hz); LF (low frequency; 0.04-0.15 Hz) and the HF (i.e., high frequency; 0.15-0.40 Hz). The HF component is coupled with the respiratory fluctuation (i.e., respiratory sinus arrhythmia) and is produced by the parasympathetic modulation on the heart. The LF is mainly coupled with variations in the blood pressure (i.e., Mayer waves), and is thought to represent the modulation of both parasympathetic and sympathetic branches on the heart. The VLF does not have a defined physiological source, but it may involve alterations in heart rate produced by hormones and body temperature (Electrophysiology, 1996).

Roughly, frequency domain analysis involves the calculation of the spectral energy content of each frequency component through a power spectral density (PSD) estimation. Several methods have been developed to perform the PSD estimation and they are generally divided into two categories that provide comparable results: non-parametric and parametric methods, each with respective pros and cons (Electrophysiology, 1996). The Welch periodogram (Welch, 1967) is a non-parametric approach based on the Fourier Transform and consists of the average of several PSD estimations on different segments of the same RRi series, which is an important approach to reduce the spectral estimation variability (Welch, 1967). On the other hand, the autoregressive technique is the most widely used parametric method to estimate the spectral components of the HRV signal (Berntson et al., 1997). The PSD estimation with the autoregressive method consists of a parametric representation of the RRi series and the frequency response of the estimated model. From the estimated PSD, generally, the following indices presented in Table 1 are calculated.

Variable	Units	Frequency Band
Total Power	ms2	0 - 0.4 Hz
VLF	ms2	< 0.04 Hz
LF	ms2	0.04 - 0.15 Hz
HF	ms2	0.15 - 0.4 Hz
LF/HF		
LFn.u	normalized units $\frac{LF}{TotalPower-VLF}$	
HFn.u	normalized units $\frac{HF}{TotalPower-VLF}$	

Non-linear indices are also frequently used to extract information from the ANS based on the heart rate fluctuations patterns. SD1 and SD2 respectively reflect the short and long term fluctuations of heart rate, which can be calculated from the Poincaré ellipse plot (Berntson et al., 1997) or derived from SDSD and SDNN values as shown by equations 2 and 3 below.

$$SD1 = \sqrt{2SDNN^2 - 2SD2^2}$$

- Equation 2

$$SD2 = \sqrt{2SDNN^2 - \frac{1}{2}SDSD^2}$$

- Equation 3

Bartels et al., (2019). HRV: a Pythonic package for Heart Rate Variability Analysis. *Journal of Open Source Software*, 4(43), 1867. https://doi.org/10.21105/joss.01867

The hrv is a simple and open-source Python module that brings the most widely used techniques to work with the RRi series and HRV analyses without losing the power and flexibility of a native Python object and a numpy array (Oliphant, 2006). The following sections present the basic workflow with RRi series, starting with reading a file containing a tachogram, visualizing the given RRi series, dealing with noise filtering and, finally, calculating time/frequency domain and non-linear HRV indices.

Basic Usage

This section presents a non-exhaustive walkthrough of the features offered by the `hrv` module. To have access to the source code and more usage examples, please refer to the [software repository](#).

Once the RRi series is created in Python using the `hrv.io` submodule, which supports text, CSV and hrm (PolarTM) files, or from any Python iterable (i.e lists, tuples, etc), an RRi instance with the necessary methods to implement the Python iterable pattern is created. With the RRi object it is possible to iterate (i.e `[r for r in rri_series]`), search for a value at a given index (i.e `rri_series[0]`), and slice the tachogram (i.e `rri_series[5:10]`). As the RRi class also implements some of the behaviors of the numpy array (Oliphant, 2006), it is possible to perform math operations with the tachogram, i.e: `rri_series / 1000`.

The RRi class also has methods for basic statistical metrics calculation, such as average, standard deviation, min and max, and others. In order to access a complete Python dictionary containing all available statistical metrics of an RRi instance, it is possible to call the `describe` method. Features for visualization are also present in the RRi class. In order to visualize the time series represented by the RRi series, the `plot` method can be called. The visualization of the histograms showing the distributions of RRi or heart rate time series is also possible with the method `hist`. '
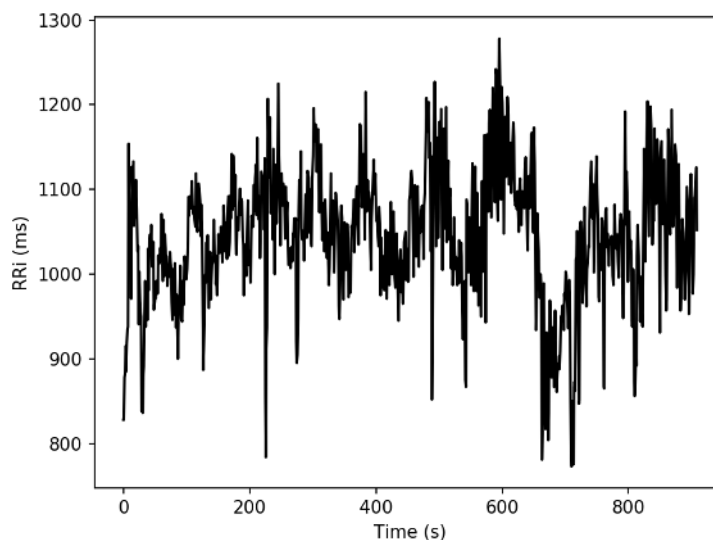
Read a file containing RRi values and visualising it

The following code snippet shows how to read a RRi series from a single column CSV file and plot the respective series with black lines.

```
from hrv.io import read_from_csv

rri = read_from_csv('path/to/file.csv')

fig, ax = rri.plot(color='k')
```

Bartels et al., (2019). HRV: a Pythonic package for Heart Rate Variability Analysis. *Journal of Open Source Software*, 4(43), 1867. https://doi.org/10.21105/joss.01867

Figure 1: RR intervals of a young subject at rest condition produced with the `plot` method from the RRi class.

To retrieve statistical properties of a RRi series the method `describe` can be invoked:

```
desc = rri.describe()
```

```
----------------------------------------
                   rri            hr
----------------------------------------
min             750.00         66.30
max             905.00         80.00
mean            805.50         74.78
var            2646.25         20.85
std              51.44          4.57
median          805.00         74.54
amplitude       155.00         13.70
```

```
print(desc['std'])
{'rri': 51.44171459039833, 'hr': 4.5662272355549725}
```

Filtering the RRi series

In some cases and for many different reasons, the tachogram may present with movement artifacts or undesired RRi values, which may jeopardize the analysis results. One way to deal with this scenario is to apply filters to the RRi series. For this reason, the hrv package offers three lowpass filters for noise removal: moving average, which given an order value N, replaces every RRi value by the average of its N neighbors values; the moving median, which works similarly to the moving average filter, but apply the median function; and the quotient filter (Piskorski & Guzik, 2005), that removes the RRi values which the ratio with its adjacent RRi are greater than 1.2 or smaller than 0.8.

```
from hrv.filters import moving_median, quotient

filt_rri_median = moving_median(rri, order=3)
filt_rri_quotient = quotient(rri)

filt_rri_median.plot(ax=ax)
filt_rri_quotient.plot(ax=ax)
```
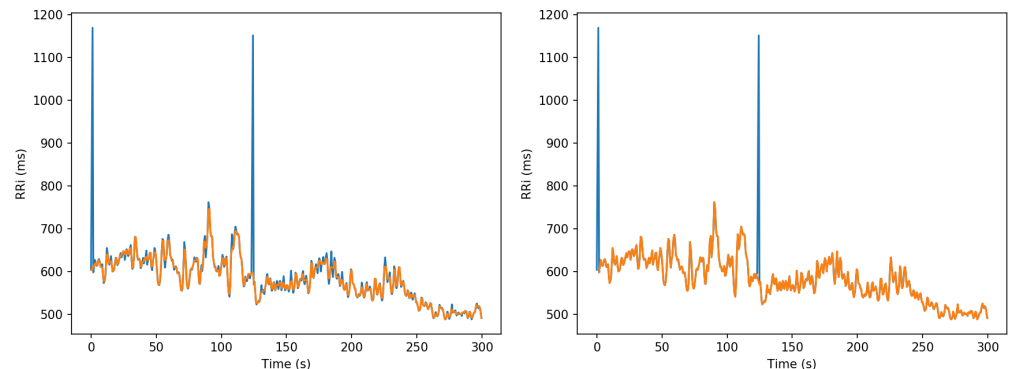

Figure 2: The left panel shows the original RRi (blue) and after filtering with a moving median filter with order equal to 3 (orange). The left panel depicts the original RRi (blue) and after filtering with a quotient filter (orange). This picture was created using the "plotmethod `from theRRi`"' instance.

Time Domain Analysis

In order to calculate the time-domain indices, the function `time_domain` can be imported from the submodule `hrv.classical` and applied to any Python iterable containing the RRi series including the RRi instance from the module presented in this article.

```
from hrv.classical import time_domain

results = time_domain(rri)
print(results)

{'mhr': 66.528130159638053,
 'mrri': 912.50302419354841,
 'nn50': 337,
 'pnn50': 33.971774193548384,
 'rmssd': 72.849900286450023,
 'sdnn': 96.990569261440797
 'sdsd': 46.233829821038042}
```

Frequency Domain Analysis

Similarly to the `time_domain` function, to calculate the frequency-domain indices, the `frequecy_domain`, which is also placed in the `hrv.classical` submodule, can be used. The `frequency_domain` function present in the `hrv` module takes care of the pre-processing steps: the detrending of the RRi series (which the default is a linear function, but can be any custom Python function), interpolation using cubic splines (also accepts linear interpolation) and resampling at a given frequency, the default is 4 Hz.

Bartels et al., (2019). HRV: a Pythonic package for Heart Rate Variability Analysis. *Journal of Open Source Software*, 4(43), 1867. https://doi.org/10.21105/joss.01867

When Welch's method is selected, a window function (default: hanning), the number of RRi values per segment and the length of superposition between adjacent segments can be chosen. When the AR method is selected, the order of the model (default 16) can be set.

The area under the curve of each frequency range in the estimated PSD is calculated using the trapezoidal method. As a default, the hrv module uses the frequencies cutoffs shown in Table 1 to limit the integration range of each frequency domain indices, however, it is possible to set the frequency range of VLF, LF, and HF in the `frequency_domain` function call.

```python
from hrv.classical import frequency_domain

results = frequency_domain(
    rri=rri,
    fs=4.0,
    method='welch',
    interp_method='cubic',
    detrend='linear'
)
print(results)

{'hf': 1874.6342520920668,
 'hfnu': 27.692517001462079,
 'lf': 4894.8271587038234,
 'lf_hf': 2.6110838171452708,
 'lfnu': 72.307482998537921,
 'total_power': 7396.0879278950533,
 'vlf': 626.62651709916258}
```
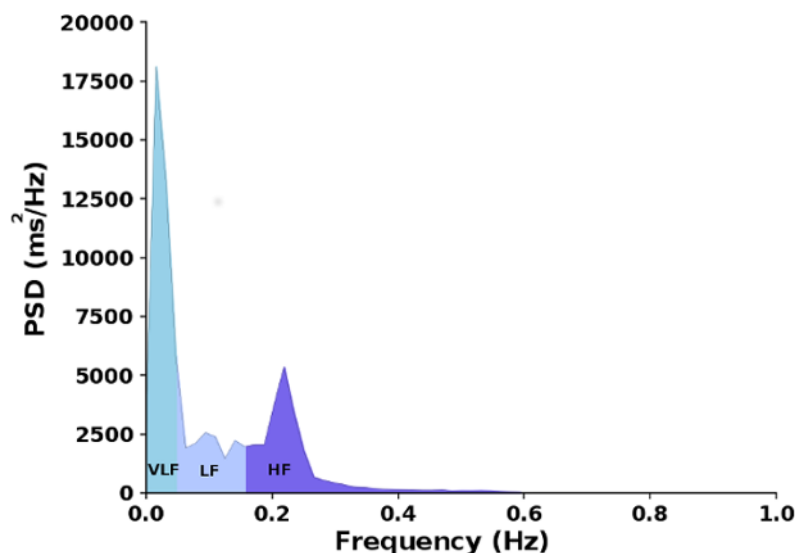

Figure 3: Power Spectral Density of a RRi series estimated with the Welch's method.

Non-linear Analysis

Finally, among the non linear metrics, `hrv` module offers SD1 and SD2, which can be calculated with the `non_linear` function from the `hrv.classical` submodule.

```
from hrv.classical import non_linear

results = non_linear(rri)
print(results)

{'sd1': 51.538501037146382,
 'sd2': 127.11460955437322}
```

Dependencies

The `hrv` package depends on the following modules: numpy (Oliphant, 2006), matplotlib (Hunter, 2007), scipy (Jones, Oliphant, Peterson, & others, 2001) and spectrum (Cokelaer & Hasch, 2017).

Acknowledgements

References

Berntson, G. G., Thomas Bigger Jr, J., Eckberg, D. L., Grossman, P., Kaufmann, P. G., Malik, M., Nagaraja, H. N., et al. (1997). Heart rate variability: Origins, methods, and interpretive caveats. *Psychophysiology*, *34*(6), 623–648.

Burton, A., Rahman, K., Kadota, Y., Lloyd, A., & Vollmer-Conna, U. (2010). Reduced heart rate variability predicts poor sleep quality in a case–control study of chronic fatigue syndrome. *Experimental brain research*, *204*(1), 71–78.

Cokelaer, T., & Hasch, J. (2017). 'Spectrum': Spectral analysis in python. *J. Open Source Software*, *2*(18), 348.

DeGiorgio, C. M., Miller, P., Meymandi, S., Chin, A., Epps, J., Gordon, S., Gornbein, J., et al. (2010). RMSSD, a measure of vagus-mediated heart rate variability, is associated with risk factors for sudep: The sudep-7 inventory. *Epilepsy & Behavior*, *19*(1), 78–81.

Electrophysiology, T. F. of the E. S. of C. the N. A. S. of P. (1996). Heart rate variability: Standards of measurement, physiological interpretation, and clinical use. *Circulation*, *93*(5), 1043–1065.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. doi:10.1109/MCSE.2007.55

Jones, E., Oliphant, T., Peterson, P., & others. (2001). SciPy: Open source scientific tools for Python. Retrieved from http://www.scipy.org/

Malik, M., & Camm, A. J. (1990). Heart rate variability. *Clinical cardiology*, *13*(8), 570–576.

Nolan, J., Batin, P. D., Andrews, R., Lindsay, S. J., Brooksby, P., Mullen, M., Baig, W., et al. (1998). Prospective study of heart rate variability and mortality in chronic heart failure:

Results of the united kingdom heart failure evaluation and assessment of risk trial (uk-heart). *Circulation*, *98*(15), 1510–1516.

Oliphant, T. (2006). NumPy: A guide to NumPy. USA: Trelgol Publishing. Retrieved from http://www.numpy.org/

Piskorski, J., & Guzik, P. (2005). Filtering poincare plots. *Computational methods in science and technology*, *11*(1), 39–48.

Sajadieh, A., Nielsen, O. W., Rasmussen, V., Hein, H. O., Abedini, S., & Hansen, J. F. (2004). Increased heart rate and reduced heart-rate variability are associated with subclinical inflammation in middle-aged and elderly subjects with no apparent heart disease. *European heart journal*, *25*(5), 363–370.

Sgoifo, A., Carnevali, L., Pico Alfonso, M. de los A., & Amore, M. (2015). Autonomic dysfunction and heart rate variability in depression. *Stress*, *18*(3), 343–352.

Weissman, A., Lowenstein, L., Peleg, A., Thaler, I., & Zimmer, E. Z. (2006). Power spectral analysis of heart rate variability during the 100-g oral glucose tolerance test in pregnant women. *Diabetes care*, *29*(3), 571–574.

Welch, P. (1967). The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on audio and electroacoustics*, *15*(2), 70–73.