# Programming Exercise 2 - MA Filters on ECGs

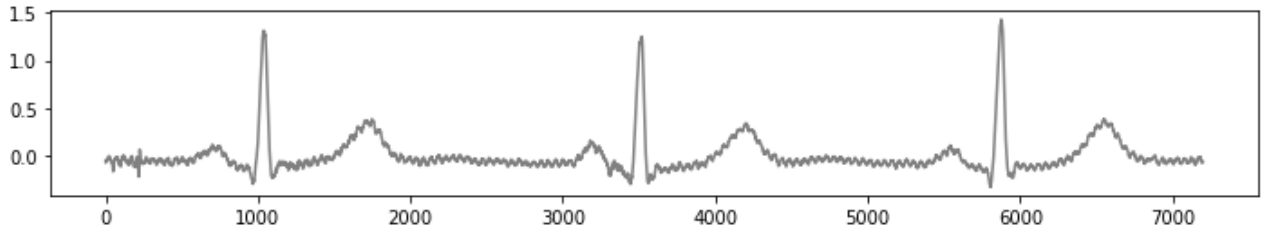
2024/11/07

**Digital Biomarkers**

R. Wildhaber

## Programming Exercise 2-1: Moving Average Filters

| | |
|---|---|
| Python script template: | `L2-Moving-Average-Filters-ECG-Template` |
| Data set: | `SECG3_FILT_HP51_3CH_20S_FS2400HZ.csv` |
| Data source: | https://lmlib.ch |




Running the template script plots the ECG signal (grey) as shown above.

The current python template applies a linear filter to an input signal $y$ which is either a real ECG signal or a synthetic rectangular signal (Use the code switch in the if-clauses in Cell (2) of the template.)

Tasks:

a) Run the full script. Cell (3) displays the unfiltered and filtered signal in a sigle plot. The default filter in Cell (2) has currently the coefficients $h = [1,]$ leading to $y[n] = x[n]$, which has no effect on the signal and needs to be replaced by you. Modify $h$ such that we get a moving average filter of any specific length $L$. Check the Python function `scipy.signal.lfilter()` for this. Tune $L$ such that the noise components in the ECG signal get suppressed while the ECG signal is only minimally modified, i.e., optimize the Signal-to-Noise (SNR) ratio.

b) After optimizing the SNR, we observe a time delay between the input signal $x$ and the filter output $y$. Modify Cell (2) to compensate for this time delay.

   Hint: Check for the Python function `np.roll(...)`.