Exercise 1. Preprocessing, artifacts and noise

Data Description: https://physionet.org/content/ecgiddb/1.0.0/

The database contains 310 ECG recordings, obtained from 90 persons. Each recording contains:

- ECG lead I, recorded for 20 seconds, digitized at 500 Hz with 12-bit resolution over a nominal ±10 mV range;
- 10 annotated beats (unaudited, i.e. not checked after automatic detection, R- and T-wave peaks annotations from an automated detector);
- information (in the .hea file for the record) containing age, gender and recording date.
- Access readme file for further information (https://physionet.org/content/ecgiddb/1.0.0/README)

Read a single file from here: https://archive.physionet.org/cgi-bin/atm/ATM

Change the settings:

- Database: ECG-ID Database (ecgiddb)
- Record: Person_01/rec_3
- Signals: all
- Annotations: unaudited R- and T-wave peaks annotations from an automated detector (atr)
- Length: 1 min
- Time format: elapsed time
- Data format: standard
- Toolbox: Export signals as csv

The output below was prepared using this command:

```
rmdsap -r ecgiddb/Person_01/rec_1 -c -H -f 0 -t 20 -v -pd >samples.csv
```

Click "samples.csv" and save the file. The file is also available in the Moodle folder of this exercise.

A sample of the raw and noise filtered ECG signals are below.
Note: If you find difficulties downloading the file, you can get it from the Moodle page

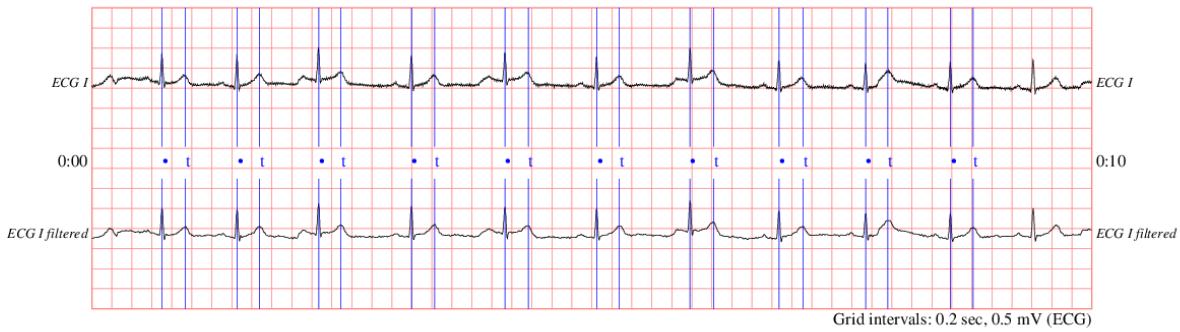

Figure 1. 10-second sample of the ECG signal for Person_01/rec_1 (Figure saved from Physiobank)

Tasks

1. Plot the unfiltered and filtered signal into same graph. Name the axes, units and signals (legends). Name the basic waveforms of the ECG signal.
   Questions: What differences can you see in the signals? What could be the source of the artefacts smoothened by filtering?

   Suggested implementation:
   - use *pandas* data manipulation (https://pandas.pydata.org/docs/user_guide/index.html)
   - use *Matplotlib* for plotting (https://matplotlib.org/stable/tutorials/introductory/pyplot.html)

2. Make another plot of the power spectral density of raw and filtered ECG signal.
   Questions: What additional information as compared to the time plot does the PSD plot give to you about the signal and possible noise?

   Suggested implementation:
   - use *plt.psd* function (https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.psd.html)
   - set the number of DFT points (NFFT parameter) to 1024

3. Detect all ECG waveforms (first the R peaks, then the other waveforms, i.e., the T-peaks, P-peaks, Q-peaks and S-peaks) for raw data. Plot the raw data indicating the detected waveforms on it. Make another delineated plot, where all cardiac cycles are on top of each other.
   Questions: Compare the detection of ECG waveforms of raw data to the filtered data. How well does the algorithm work for unfiltered data, what is the reason for this? Which waveforms are easy to detect correctly, which are hard, and why?

   Suggested implementation:
   - use *NeuroKit2* package (https://neuropsychology.github.io/NeuroKit/)
   - plot the results with *nk.events_plot* (https://neuropsychology.github.io/NeuroKit/functions/events.html#events-plot)
   - plot the delineated version of the results with *nk.ecg_delineate* (https://neuropsychology.github.io/NeuroKit/functions/ecg.html#ecg-delineate)

4. Build a simple set of filters to try to increase the quality of the unfiltered signal. Try at least the following combination (implement both filters back to back): (i) bandpass filter for 0.01 – 150 Hz, and (ii) notch filter for 50 Hz. First, plot the raw signal and its PSD. Then, design and apply the first filter and plot the filtered signal and PSD. Thirdly, design and apply the second filter and plot the filtered signal and the PSD.
   Questions: What is the idea behind the above-mentioned filters? How does filtering change the signal? Did the filtering improve the quality of the signal sufficiently? If these filters are not sufficient, what other filters could be used?

   Suggested implementation:
   - use scipy.signal to build filters (https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.butter.html and https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.iirnotch.html)