

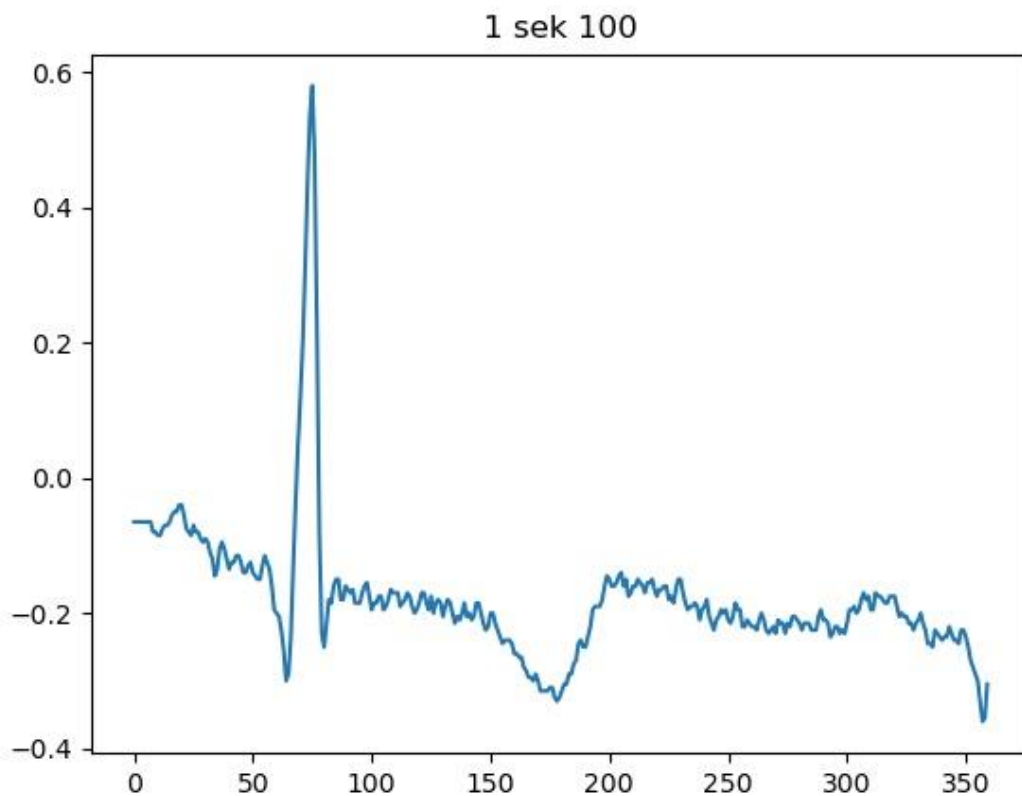
Łukasz Obłąk

Komunikacja człowiek – maszyna

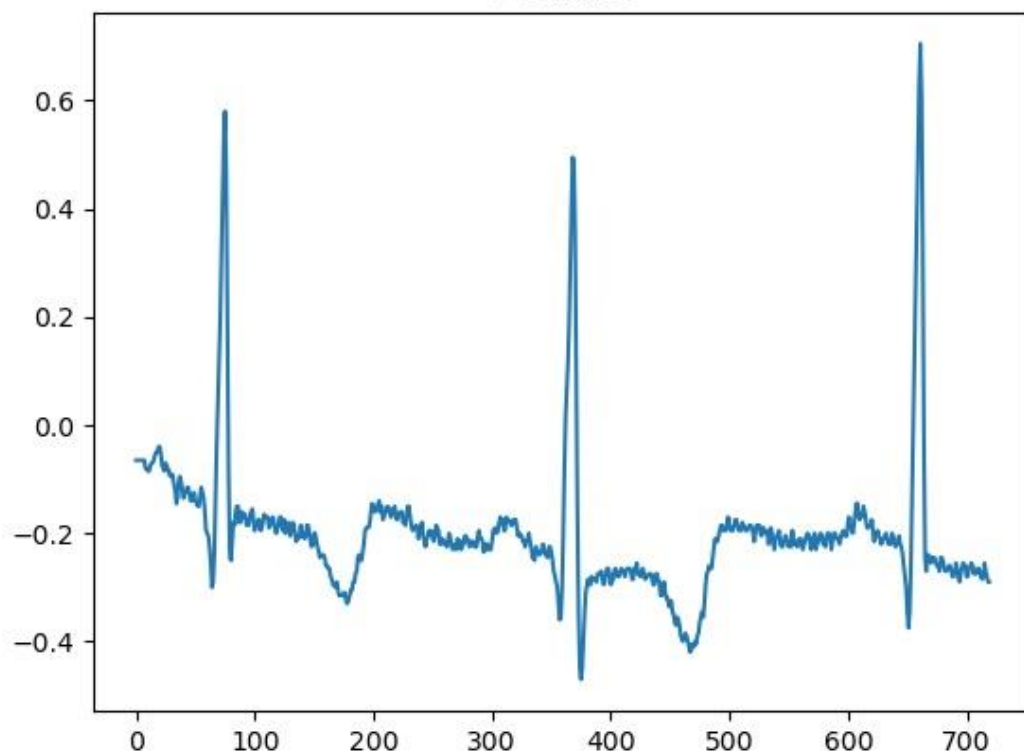
Lab 1 – EKG

Dla podanych plików zawierających przebiegi rytmu serca dla pacjentów 100 i 102 należało przeprowadzić analizę wartości maksymalnych przebiegów (załamki rytmu serca).

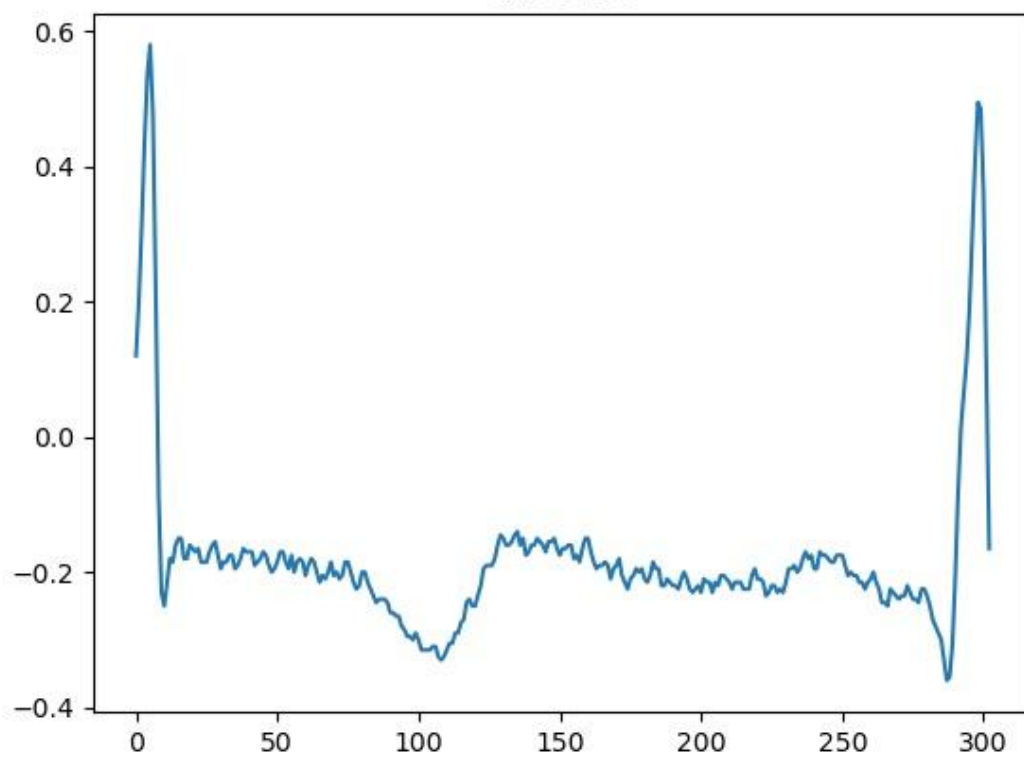
Dla pacjentów 100 i 102 wykresy przedstawiające 1 sekundę sygnału, 2 sekundy oraz jeden pełny okres między dwoma załamkami:



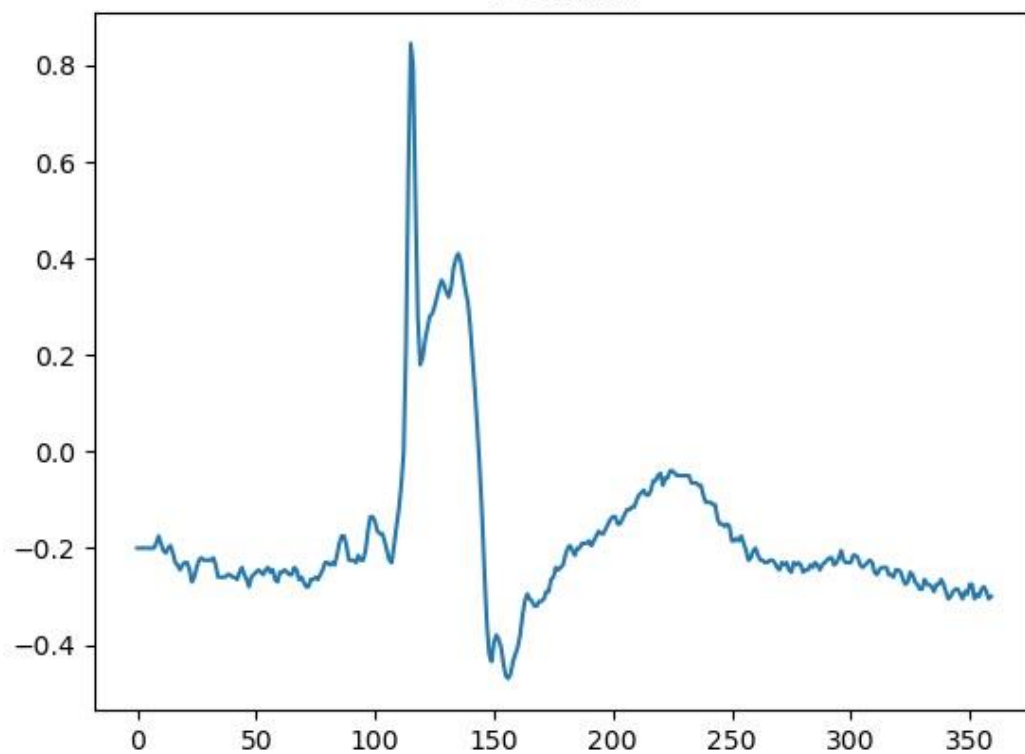
2 sek 100



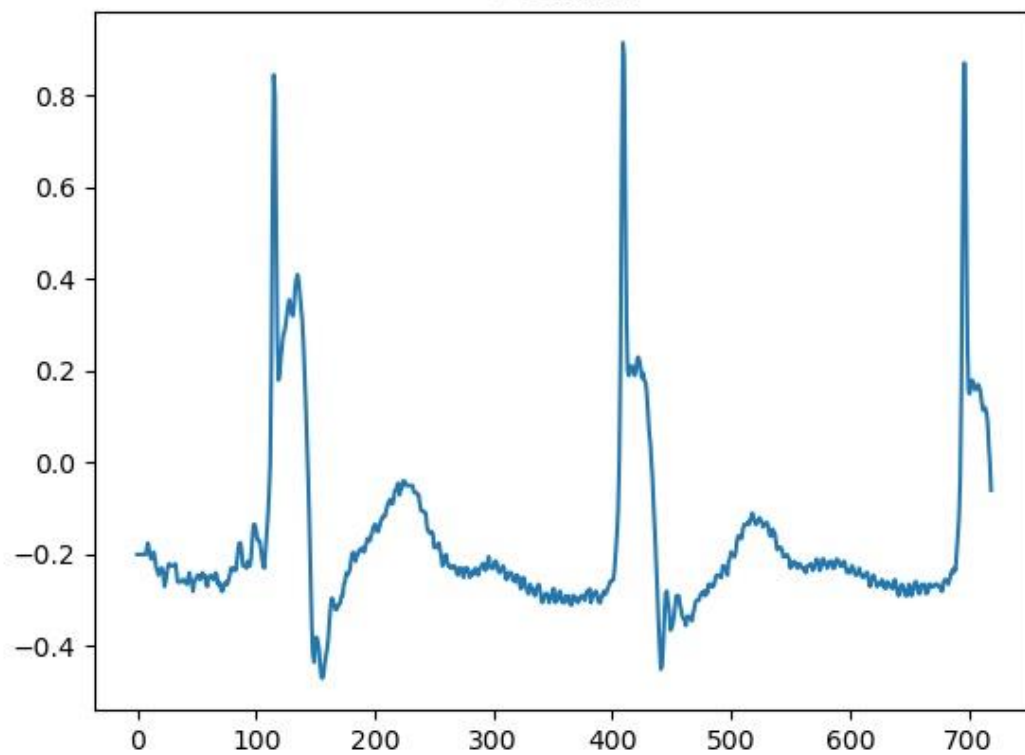
okres 100

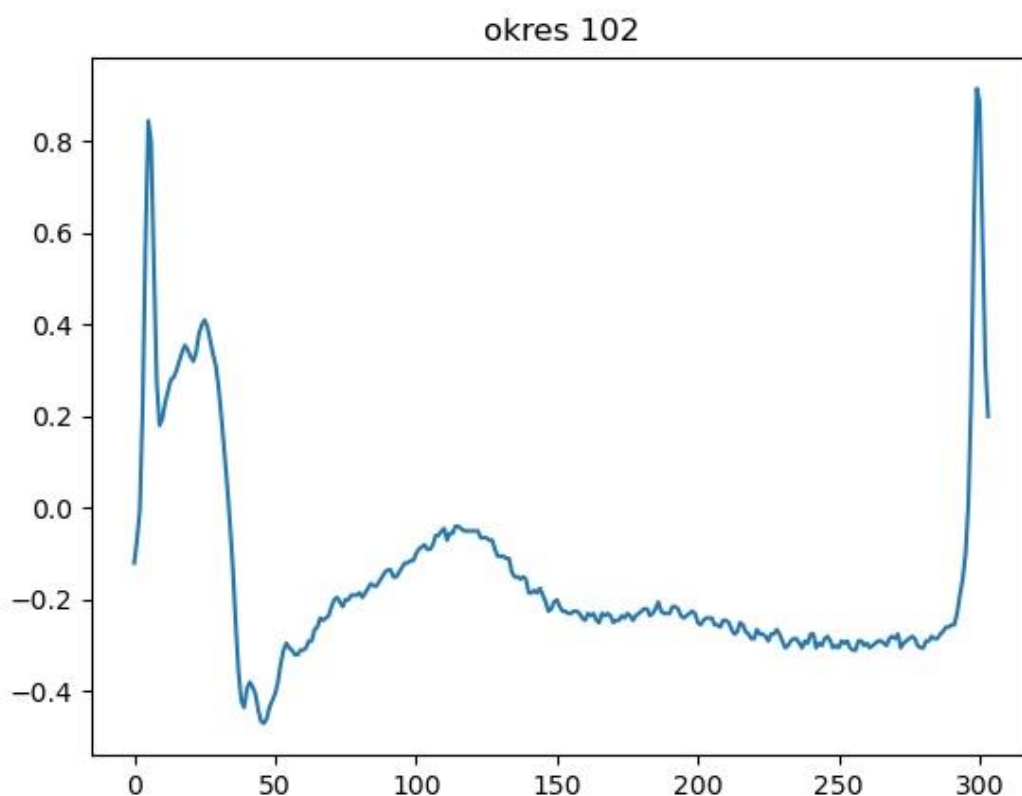


1 sek 102



2 sek 102





Nazwa analizowanego pliku	Częstotliwość próbkowania pomiarów (informacja z pliku txt)	Rytm serca na podstawie pierwszych 2ch pików	MEAN Uśredniony rytm serca	STD	VAR
100_V5.dat	360 Hz	73.720	69.385	126.944	16115
102_V5.dat	360 Hz	73.469	74.669	110.657	12245

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import find_peaks
from statistics import variance, stdev
HZ = 360

def window_indices(seq, n=2):
    # generator okienek o n długości dla danego ciągu:
    # seq = [1, 2, 3, 4]
    # assert(list(window_indices(seq, 2)), [(1,2), (2,3), (3,4)])

    i = 0
    while i+1 < len(seq):
        yield seq[i:i+n]
        i += 1
```

```

def distances(seq):
    # dla listy n peaków produkuje generator n-1 dystansów pomiędzy
    # peakami
    for i, j in window_indices(seq, 2):
        yield j-i

for patient_id in ('100', '102'):
    # wczytanie danych
    data = np.genfromtxt(f'{patient_id}_V5.dat',
                        dtype=float,
                        delimiter='\n')

    # prezentacja całości
    plt.plot(data)
    plt.show()

    # prezentacja 1 sekundy
    plt.plot(data[:1*HZ])
    plt.title(f'1 sek {patient_id}')
    plt.savefig(f'1sek_{patient_id}_V5.jpg')
    plt.close()

    # prezentacja 2 sekund
    plt.plot(data[:2*HZ])
    plt.title(f'2 sek {patient_id}')
    plt.savefig(f'2sek_{patient_id}_V5.jpg')
    plt.close()

    # znalezienie peaków dla 4 sekund
    indices, heights = find_peaks(data[:4*HZ], 0.42)

    # prezentacja 1 okresu
    margin = 5
    a = indices[0]
    b = indices[1]
    plt.plot(data[a-margin:b + margin])
    plt.title(f'okres {patient_id}')
    plt.savefig(f'okres_{patient_id}_V5.jpg')
    plt.close()

    # obliczenie rytmu bicia serca z 2 pików
    beats_per_min = (HZ / (b - a)) * 60
    print(f'bpm for {patient_id}: {beats_per_min}')

    # pokazanie 4 sekund wraz z peakami
    peak_heights = [data[indice] for indice in indices]

```

```
plt.plot(data[:4*HZ])
plt.scatter(indices, peak_heights, c='red')
plt.show()

# obliczenie peaków całego przebiegu
all_peaks_indices, _ = find_peaks(data, 0.42)

# średnie tempo bicia serca, wariancja, odchylenie standardowe
print(f'mean bpm: {(len(all_peaks_indices)/len(data)) * 60 * 360}')

print(
    f'variance for patient {patient_id}:
{variance(distances(all_peaks_indices))}')

print(
    f'stdev: for patient {patient_id}:
{stdev(distances(all_peaks_indices))}')
```