This document is formed to explain the use of bowel sound analysis script.

1. File Naming: Name the sample recordings according to the format: date-condition-order-user.
   1. Date: Use date format as YYYYMMDD (i.e. 20220129 for 29th of January, 2022)
   2. Condition: For the measurements before/after stimulation, use the format “before-stim” or “after-stim”. For those during stimulation, use “frequency-duration” such as “20Hz-20dk”.
   3. Order: The order according to condition. If it is the 5th measurement during stimulation, insert 5.
   4. User: Use initials.

Examples:

*20220129-before-stim-1-tt.wav*

*20220129-20Hz-20dk-4-tt.wav*

*20220129-after-stim-2-tt.wav*

1. How to use the script:
   1. The script takes only .wav files as input by default. If any other sound files are to be analyzed, change line 23 accordingly.
   2. For exporting the analysis results to a .csv file, set the write\_switch to 1.
   3. The script exports the results to a file named *data.csv* by default. If there is no such file, it creates the file. If there is, it adds the results to the end of the result file. If you wish to modify, change the lines 19&128 accordingly.
   4. The data are written in the .csv code row by row, one row spared for each sound file. In each row, the parameters are delimited by space. This can be modified in line 125. Be careful not to use comma as delimiter, as it might cause complication in analyzing decimal numbers.
   5. The parameter “order\_score” is a dummy value used for ordering the files chronologically in the output file (*data.csv* by default). In the .csv file, use Excel to order the rows according to this score.
2. How bowel sounds are detected:
   1. The bowel sounds are searched within the original signal between lines 59-76 in the script.

max\_array=[]

max\_index\_array=[]

bowel\_array=[]

bowel\_array=np.array(bowel\_array)

for i in range(1,length\_audio,200):

if np.amax(audio\_in[i:i+198])>100 and np.amax(audio\_in[i:i+198])!=audio\_in[i+198]:

cpm+=1

max\_index = np.argmax(audio\_in[i:i+199])

m = np.amax(audio\_in[i:i+199])

max\_index = i+max\_index;

max\_array.append(m)

max\_index\_array.append(max\_index)

if max\_index>29 and max\_index<479931:

bowel\_array = np.concatenate((bowel\_array, audio\_in[max\_index-30:max\_index+70]))

#audio\_in[max\_index-30:max\_index+70]=0

bowel\_duration=(audio\_duration\*len(bowel\_array))/length\_audio

* 1. In this script, the for loop searches for bowel sound moments by checking whether the signal intensity exceeds “100” (a threshold determined based on the data, in such a way that the detectable bowel sounds are always above 100 at the peak). But this search is performed by checking 200 samples at a time (hence the increment of 200 in for loop). This is just to make sure each bowel sound moment is taken into account only once, since the oscillation may exceed 100 in amplitude more than once.
  2. When a bowel sound moment is detected, the parameter *cpm* (counts per minute) is incremented by 1.

NOTE: Even though it is a parameter of clinical use, it is not so reliable. This is because when the bowel sounds are connected (the next one starts immediately after the previous one finishes), the whole sequence is counted only once.

* 1. Then, these bowel sound moments are trimmed from the original signal and concatenated to *bowel\_array* signal. This becomes a sound signal formed only by the bowel sounds, getting rid of the silent moments.
  2. The length of the bowel signal, represented by *bowel\_duration* is one parameter to look for. It is determined by multiplying the sampling period (given by *audio\_duration/length\_audio)* with the length of the bowel array.
  3. Bowel duration is a parameter worthy of checking because it determines basically how often the bowel sounds occur.
  4. The second parameter to check is about the intensity of the bowel signals. It is calculated in lines 103-104.
  5. The parameter *total\_area* is just the summation of the absolute bowel array (absolute because the sound signal oscillates around 0). In order to make it comparable to the results obtained earlier using MATLAB, a conversion ratio is added. This can be removed in the future uses.

total\_area = sum(abs(bowel\_array))/100\*0.003 #For conversion to the values obtained in MATLAB

avg\_area = total\_area/bowel\_duration

* 1. The third and final parameter to check is *avg­\_area.* This basically determines how strong each bowel sound is produced. It is calculated by ratioing the *total\_area* and *bowel\_duration.*