INTRODUCTION

This is a workflow for:
- Tracing respiration curves based on detected temperature changes in the nasal area of a subject.
- Detecting the three main breathing phases (inspiration, expiration and expirational pause)
- Measuring three main parameters (duration, amplitude, slope) of each breathing phase
- Synchronising respirational trace with the timing of vocal emission
- Marking breathing movements associated with vocal emission
- Marking breathing movements associated with non-focal calls
- Performing permutation procedure for testing the vocalisation effects on focal respiration
- Performing permutation procedure for testing effects of non-focal vocal signals on the respiration of the focal individual
- Plotting

The process is divided between two main script files:
   The merging of input data and detection of respiration traces is achieved by nostril_apply_script.R script.
   Permutation procedures, calculation of pseudo P values and plotting is done by permutation_script.R.

REQUIREMENTS

The process relies on inputs from several sources (names match to names of the data folders:
- imageJ_output: per-frame temperature estimations at the nasal region of interest
- loopy_output: coordinated of nostril ROI tracking and information on laterality of each detection point
- outputs: start and end times of produced calls with complementary metadata on caller ID and date of recording.
                  Audio/Video log file with call times to video frames conversion info.

Script file: nostril_apply_script.R

Data loading and merging

The initial steps require merging of data files from multiple sources using different naming conventions. The script is not generalised and works with the naming conventions used in the specific project. The provided sample files demonstrate the process of merging and synchronising the data however usage with files of different formats, structure and naming scheme will require adaptation of the corresponding parts of the code.

- After listing and reading in the initial data files a conversion table (proc_file_list) is made between imageJ_output files and loopy_output. The matching is achieved on the name of the video file used to generate them.
- The per-frame intensity readings are assigned to Left or Right nostril, based on the nasal tracking results.
- Intensity is transformed to temperature using raw2temp function and by providing a list of calibration constants from FLIR.csq files. Similar functionality is available in the Thermimage package
-

The data is going through several quality control steps:
- Sequences with above 5 frames gaps are omitted
- Sequences below 100 frames length are omitted
- Temperature spikes of above 1.5 degrees are omitted
- Unique segment IDs are assigned
- Sequences in which both Left and Right nostrils are tracked are kept
- Temperature curves are plotted and visually inspected for cyclic patterns
- The segments are manually selected for the nest data processing steps

Synching of call times with video frames:
- Full call times csv is loaded "thermal_audio_raw_timing.csv"
- The recording metadata is arranged

- Logfile listing Audio-Video name conversion and frame sync data is loaded "thermal_logfile.csv"
- Sync time between calls and video frames is calculated and object AVsummary is created including the Region Of Interest thermal measurements, marked frames associated with focal calls (1) and well as frames associated with non-focal calls (0.5)
- Respiration curves are plotted for another visual inspection

Digital filter data smoothing:
- Temperature values are cantered to avoid edge artefact of the filter
- Butterworth low pass filter is applied to smooth low frequency noise. The settings of the filter are customisable and should depend on the data
- The filtered and non-filtered temperature trace is plotted for visual inspection

Breathing phase detection:
- A noise threshold value is determined by trial and error (noise.thresh <- 0.25)
- Local min and max points are detected by crawling on the temperature curve
- The process is auto-corrected by flexibly adapting the noise.thresh value according to the calculated breathing rate, keeping it within species dependent biologically relevant range
- An optional manual step of peak corrections can be done, based on visual inspection of the breathing curves and min and max locations
- A secondary detection of expiratory pause phase is done by pulling the min points and crawling along the curve towards the points of curve undulation.
- A second optional step of manual correction of the respiration phase transition points can be done based on visual inspection of the breathing curves

Respiration phase measurements:
- Three metrics are extracted for each of the respirational phases. Duration, Amplitude (change in temperature), Slope (change in temperature/duration)
- Based on the timing of the focal calls, respirational cycles are marked as Call(c), Befor call (bc), after call(ac), and quiet(n).
- Based on the timing of non-focal calls, breathing phases during which or immediately preceding which non-focal calls were detected are marked as "y"
- Respiration rate per segment is calculated for internal control

Data is collected and summarized in a summ object
Output files are generated – "peaksummary4stats.csv" containing only the respiration phase measurements and "full_summary4fig.csv" containing all the temperature points with marked vocalisation associated frames.

Script file: permutation_script.R

REQUIREMENTS

This stage requires the input of data_summary.csv. A summary file detailing the calculated measurements (amplitude, duration, slope) of respiration phases (inspiration, expiration, expiratory pause) and the their definition as Quiet or Call associated (Pre call, Call, Post call).

Control for breathing symmetry, individual IDs and mark Non-focal calls associated respiration:
- Calculating the proportion between expiration and inspiration phases to control for approximate symmetry
- Generating individual IDs for each caller
- Mark respirational phases synchronised or immediately following Non-focal vocalisation
- Filter data to include individuals that emitted calls, to allow permutation procedure (object: all_peaks_callers)
- Break the dataset according to the three breathing phases (objects: Phase_1, Phase_2, Phase_3)

Permutation procedure:
- Set number of permutation rounds in - perm_rounds
- run permutation tests with individual ID as a blocking parameter for within individual restricted permutation. The comparisons are made for each one of the

respiration phases, between Quiet and one of the Vocalisation associated phases (Pre call, Call, Post call).
- For each permutation round a Mean difference is calculated and stored.
- A histogram of permutation generated differences is plotted with the data value marked in red.
- A pseudo P-value is calculated for each pairwise comparison by dividing the number of permuted values that are more extreme then the data driven value and dividing by the total number of permutations.

```
# the permutation procedure is than repeated to assess the effect of Non-focal
calling on Focal respiration. For this test only the Quiet respiration cycles are
used (object: all_peaks_n).
```

Plotting:
- The current script generates a series of histograms and box plots. The plotting code is not fully optimized and should be further adapted to the specific needs od the user.