March 17th, 2023

**RippleDetector\_class (Shdema Epstein and Maya Geva-Sagiv)**

The ripple detector class includes several functions:

1. Detects ripples in the data based on the method described in Staresina et al 2015.
2. Plot ripple detection

**A. Detection:**

detectRipples

The method receives as input: the data, and a vector of sleep scoring (optional), IIS times (optional). The sleep scoring is a vector in which each element represents the sleep stage for an epoch of the duration stored in the property sleepEpochs (e.g. 1 second). The method will detect ripples only in the sleep stages specified by the property sleepEpochs (in Staresina et al: they detect only in stages 2-4 of the NREM). IIS times is a vector with the times of interictal spikes (for example the output of SpikeWaveDetector).

The method performs the following steps:

* Filters the data using a bandpass filter in the range specified by the properties minFreq and maxFreq (by default, as in Staresina et al, 80-100 Hz).
* Changes all the points in the data which do not belong to the required sleep stages (which are stored in the property sleepEpochs) to NaN.
* Removes IIS – turns to NaN windowAroundIIS ms before and after each IIS (by default 500 ms).
* Calculates the root mean square of the filtered data over a moving window of RMSWindowDuration milliseconds (by default, 20 ms).
* Calculates a threshold which is the rippleThreshPercentile of the calculated r.m.s. signal (by default, the 99th percentile).
* Finds segments of at least minDurationAboveThresh milliseconds which pass the threshold (by default 38 ms). These are the proposed ripples.
* Goes over the proposed ripples, for each proposed ripple:
  + Check whether more than minPercZerosAllowed (0.1 by default) of the ripple segment is zero, in which case it is excluded (note: this is an addition to Staresina’s method I added to deal with the segments of zeros in our data).
  + Smooths the original raw data in the proposed ripple segment using an averaging filter of 3 consecutive points.
  + Finds local minima and maxima in the smoothed segment, a ripple is verified only if there are more than minNumOfExtreme minima or minNumOfExtreme maxima.
  + The index of the ripple is set at its highest peak.

plotRipples

Plot the single ripples. Receives as input the data and the ripple times. The plots contain both the original raw signal and the filtered signal, the ripple time is exactly at the middle of each plot (by default 0.5 second is presented before and after, so the ripple index is at point 500 ms. The presented window duration can be set by the property secondBefAfter). The size of the subplots is set in the properties subplotSizeX and subplotSizeY.

If the parameter folderToSave is provided the method will save the plots into the folder (and not present them on screen).

PlotRipplesBiPolar

The method runs ripple detection on channels with bipolar referencing and saves figures with the results. The method goes over the required channels, references them to the provided reference, runs ripple detection on the referenced data, and plots single ripples into the required folder. Note: the spike times which are taken into account when detecting (i.e. – no ripples will be detected in the vicinity of a spike) are the unity of spikes from both channels.

Input:

runData –

A struct array with the length as the number of patients. Each element (=patient) should have the fields:

biPolarCouples – a matrix with the size <# of chans>\*2. Element <i, 1> includes the channel index on which we want to detect ripples, element <i, 2> includes the channel to which it is referenced. i.e. the detection is run on data\_<i,1> - data\_<i, 2>.

DataFolder – The folder in which the raw data is stored. The property fileNamePrefix of the class includes the prefix for the data filenames (by default: ‘CSC’).

SpikesFileNames – The filenames (including path) of the mat files that include the spike times. The method assumes the filename format is SpikesFileNames<#chan index>.

folderToSave – folder name into which to save the plots of the detected ripples. The method uses plotRipples to create the figures and the figures’ properties are set by the same properties as the ones setting plotRipples’ output.

Comparison Between Ripple Detection Criteria in Various Publications:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Paper** | **Bandpass range** | **Data on which detection was performed** | **Ripple criterion** | **Additional Conditions** | **Electrode Referencing** |
| Staresina, Nature Neuroscience 2015 | 80-100 Hz | Artifact-free NREM stages 2-4 | >99% percentile of average of 20 ms of RMS, passed for >= 38 ms | At least 3 peaks or 3 troughs in the raw signal | Mastoids. Analysis was performed also after re-referencing the hippocampal data to another contact on the same depth electrode with the same pattern of results. |
| Staba, American Neurological Association, 2004 | 80-500 Hz (detects ripples and fast ripples together) | Wake+NREM+REM | Successive RMS amplitude greater than 5 stds of the mean (it’s not mentioned what defines “successive”) |  | Contra-lateral auricule site |
| Zhang, Nature Communications 2018 | 80-100 Hz (note: they tried also 80-140 Hz and found 95% of the ripples were in the 80-100 Hz range) | Artifact free data (IIS – removed using manual and automatic detection) | >99% percentile of average of 20 ms of RMS, passed for >= 38 ms | At least 3 peaks or 3 troughs in the raw signal | Mastoids, re-reference before analysis to the average across all depth and subdural electrodes |
| Vaz, Science 2019 | * 1. Hz | Data after removal of IIS | >2 stds of the amplitude of the Hilbert transform, for at least 25 ms, with max amplitude >3 stds | Adjacent ripples less than 15 ms apart were merged | Local – bipolar referencing, difference between pairs of adjacent electrodes |