**SpindleDetector**

The SpindleDetectorClass.m implements two methods:

**Method 1**

The method described in ‘Sleep Spindles in Humans: Insights from Intracranial EEG and Unit Recordings' Andrillon, Nir, et al, j of neuroscience, 2011.

The class implements 3 steps in the spindle detection process:

1. Verification of the channel (verifyChannelStep1)
2. Spindles detection (detectSpindles)
3. Second verification of the channel given the detected spindles (verifyChannelStep3)

Step 1 – verifyChannelStep1

The method performs the following steps:

* If a sleep scoring (input parameter sleepScoring) is provided, leaves only the segments of sleep in the data.
* Divides the data to segments (the segments’ length is set by the property segmentsLengthStep1)
* For each segment, if it doesn’t contain too many NaN values (i.e. not enough sleep data, threshold is set by the property nanThresh):
  + Calculates the power spectrum within the frequency range minFitRange to maxFitRange (the default is 3-30).
  + Fits the model ax^b+c to the power spectrum (the property restrictFitPowerToNegative controls whether b is restricted to negative values, and by default is true).
  + Evaluates the values of the fitted function for the frequency range (i.e. the “approximated smooth spectrum”).
* Calculates the mean of all the true power spectrums (‘true mean’) and the mean of all the evaluated power spectrums (‘evaluated mean’).
* Finds the frequency for which the difference between the true mean and the evaluated mean is maximal, within the bounds of the spindle frequencies (set by spindleRangeMin and spindleRangeMax, by default 9-16).
* Performs a one tailed t-test for the difference between the true power spectrums and the evaluated power spectrums at the frequency of maximal difference calculated at the previous step (testing the hypothesis that the true power spectrums are larger than the evaluated power spectrums at this frequency).
* If the p-value of the t-test is smaller than a threshold (pvalThreshStep1) the channel is verified.

Step 2 – Spindles Detection (detectSpindles)

The method performs the following steps:

* If a sleep scoring (input parameter sleepScoring) is provided, leaves only the segments of sleep in the data.
* Performs a bandpass in the spindles range (set by spindleRangeMin and spindleRangeMax, by default 9-16).
* Performs a hilbert transform on the bandpassed (spindle-range) data and calculates its envelope (‘spindles range data’).
* Performs a bandpass in the rejection range, i.e. the range in which if there’s high energy it means it’s not a true spindle (set by rejectionRangeMin and rejectionRangeMax, by default 20-30).
* Performs a hilbert transform on the bandpassed (rejection-range) data and calculates its envelope (‘rejection range data’).
* Finds points in which the spindle range data is at least detectionThresholdSD STDs above the mean and the rejection range data is below detectionThresholdRejectionSD STDs above the mean (and according to the provided sleep scoring it’s a moment of sleep). Spindles are sequences which have peaks that satisfy this condition (‘spindle peak’) and are of a duration within the required duration limits.
* For calculating the duration of the sequence the method finds the start and end points of the spindle sequence:
  + The start of the spindle sequences is defined as the last point before the spindle peak for which the spindle range data was more than detectionThresholdStartEndSD above the mean **and** one point before was below the threshold.
  + The end of the spindle sequence is defined the first point after the peak for which the spindle range data was more than detectionThresholdStartEndSD above the mean **and**  one point after was below the threshold.
  + The duration is the time that passes between the start and the end points, and it should be longer than eventMinDuration and shorter than eventMaxDuration (by default 0.5-2 seconds) for the sequence to be considered a true spindle sequence.
* Merges spindle sequences which are too close together (the difference between the end point of one to the start point of the next one is smaller than minDistBetweenEvents).
* Sets the spindle times to be the point of maximal power of bandpassed signal.
* If the input parameter returnStats is true it also returns extra info for each spindle:
  + startTime, endTime, duration - Start time, end time and the duration of the sequence.
  + peakTime, peakEnergy - index (time) of the peak of the spindle sequence (i.e. of the spindle range data) and the value at the peak.
  + freqSpindle, sigmaPower - the frequency with the maximal energy for the spindle sequence (calculated by first calculating the power spectrum for the spindle frequency range) and the total sum of the power spectrum over the spindle frequency range for the current spindle sequence.
  + currentStage - the current sleep stage (-1 if mixed).

Step 3 - verifyChannelStep3

The method performs the following steps:

* For each detected spindle
  + Calculates the power spectrum for a time window around the spindle (the window size is set by timeWindowAroundSpindle, and by default is 1 sec).
  + Finds N (= the property controlsPerSpindle) random control segments in the vicinity of the spindle (min distance from the spindle – minDistControlSpindle, by default 2 seconds, max distance from the spindle – maxDistControlSpindle, by default 5 seconds). The segments’ length is also set by timeWindowAroundSpindle (by default, 1 second). Calculates the power spectrum for each control segment.
* Calculates the mean of all the spindle power spectrums (‘spindle mean’) and the mean of all the control power spectrums (‘control mean’).
* Finds the frequency for which the difference between the spindle mean and the control mean is maximal, within the bounds of the spindle frequencies (set by spindleRangeMin and spindleRangeMax, by default 9-16).
* Performs a one tailed t-test for the difference between the spindle power spectrums and the control power spectrums at the frequency of maximal difference calculated at the previous step (testing the hypothesis that the spindle power spectrums are larger than the control power spectrums at this frequency).
* If the p-value of the t-test is smaller than a threshold (pvalThreshStep3) the channel is verified.

**Method 2 -** detectSpindlesStaresina

Based on Staresina 2015 with a minor modification of additional rejection band criteria.

The method performs the following step:

* If sleep scoring is provided – sets all non NREM sleep points to NaN.
* If IIS times are provided – sets the vicinity of IIS to NaN (+- 500 ms).
* Filters the data to spindles range (by default – 9-16Hz).
* Filters the data rejection range (by default – 20-30 Hz).
* Calculates the envelope of the Hilbert transform in the rejection range. Finds points in the zscored envelope which pass a rejection threshold (by default 5).
* Calculates the root mean squared signal (RMS) of the filtered (spindles range) data in a window of length RMSWindowDuration (by default 200 ms).
* Finds the RMS points which are at the top detectionThresholdPerc (by default top 25%).
* Finds segments which pass the threshould for longer than eventMinDurationStar and shorter than eventMaxDurationStar (by default 0.5-3 sec).
* Merges segments which are close together (less than minDistBetweenSpindles, by default 1 sec)
* Remove segments where any of the data points is in the rejection range.
* The spindle times are the indices of the max value per spindle.

plotSpindles

Receives as input parameters the data, the spindle times, and the number of spindles to plot per figure (by default 1), and plots each spindle and its spectrogram (the window size to plot is set by plotBeforeAfter, 500 ms before and 500 ms after by default).

plotSpindlesSimple

Plots single spindles. The number of spindles per figure is defined by subplotSizeX and subplotSizeY. Saves the plots to the folder folderToSave if provided.