PROJECT 1

INTRODUCTION

The problem proposed for this project was to analyse the sleep spindles in COVID-19 survivors and compare them with the sleep spindles in a subject who has never been infected by the virus.

We know from literature that sleep spindles are bursts of neural oscillatory activity that are generated during stage 2 NREM sleep in a frequency range of ~11 to 16 Hz (usually 12–14 Hz) with a duration of 0.5 seconds or greater (usually 0.5–1.5 seconds). Research supports that spindles play an essential role in both sensory processing and long term memory consolidation.

The aim of the analysis was to analyse the sleep spindles to get informations about the symptoms induced by COVID-19 affection and if the stress caused by the disease could contribute to cognitive ipairment.

METHODS

The work was done through the following steps:

1) Data loading

   First of all we need to load the data from CTRL033_nap.mat and ICU023_nap.mat, which contains the EEG data acquired during a nap for respectively the control subject (a participant who has never been infected by the virus) and a participant discharged from an ICU (Intensive Care Unit) due to COVID-19 infection.
   Then the time vectors were created for both the EEG signals, considering that the sampling rate is 250 Hz.

2) Filtering

   Both the signals were filtered firstly in the slow tracks [9-12] Hz, and secondly in the fast tracks [12-16] Hz. The filtering was done by applying in sequence a high pass filter and a low pass filter in order to create a band pass filter.
   The transfer function of the filter was obtained by the MatLab function butter with order 4 and the filtering was made through the MatLab function filtfilt.

3) Spindles identification

   To identify the spindles we need to define first the width of the spindle window, which can be calculated by the moltiplication of the duration of one spindle (500 msec) for the sampling rate (250 Hz).

Time instants are conteined in spindles_timing_033.mat and spindles_timing_023.mat, so these data have to be loaded in the workspace.
In order to create the 3D matrices we made use of the struct. The code below is related to the identification of the slow spindles for the CTRL subject.

```
ctrl_slow_spindles = {}; % Inizializzazione struct

%Selezionare le porzioni di segnale relative agli spindle
for spindle=1:length(ctrl_slow_timing)
    start_sample = round(ctrl_slow_timing(spindle,1)); % Campione iniziale
    end_sample = start_sample + spindle_wnd - 1; % Campione finale
    ctrl_slow_spindles{end + 1} = ctrl_slow(start_sample:end_sample,:);
end
CTRL_SLOW_SPINDLES = cat(3, ctrl_slow_spindles{:}); % Concatenzazione 3D
```

The same method was used to identify the fast spindles.
Then the whole procedure was repeated for the spindles identification of the ICU data.

4) Channels averaging

After the previous steps, we computed the average of channels for every spindle.
The dimensions of the resulting matrices are Spindle length * 1 * Number of spindles .
For instance, the code below shows the channel averaging of the CTRL slow spindles matrix.

```
CTRL_SLOW_SPINDLES_AVG = zeros(size(CTRL_SLOW_SPINDLES,1), ...
1,size(CTRL_SLOW_SPINDLES,3)); % Inizializzo la matrice

spindle_count = size(CTRL_SLOW_SPINDLES,3); % Numero di spindle

for spindle=1:spindle_count
    data_selection = CTRL_SLOW_SPINDLES(:,:,spindle); % Singolo trial
    channels = size(CTRL_SLOW_SPINDLES,2); % Numero canali del trial
    result = zeros(size(CTRL_SLOW_SPINDLES,1),1); % Risultato temporaneo
    for chn=1:channels
        result = result + data_selection(:,chn);
    end
    result = result ./ channels; % Calcolo della media lungo i canali
    CTRL_SLOW_SPINDLES_AVG(:,1,spindle) = result;
end
```

First of all, the new matrix of the averages is created with the predicted dimensions. For the single spindle, we estract the data from the complete matrix and we sum the values of every channel, then we divide the resulting vector for the number of the channels. This process is repeated for every spindle and the results are put into the matrix of the averages.
This process is repeated for the CTRL fast spindles and also for the ICU slow and fast spindles.

5) Spectra computation & 6) Spectra selection

As we did in step number 3, we made use of the struct in order to create the 3D matrices for the spectra.
The following code is related to the slow spindles spectra computation and selection for the CTRL subject. However, the process is exactly the same for fast spindles and it's repeated for the ICU subject.

```
ctrl_slow_spindle_spectra = {}; % Inizializzazione Struct
ctrl_slow_spectra_select  = {}; % Inizializzazione struct degli spettri
                                % ammissibili

spindle_count = size(CTRL_SLOW_SPINDLES_AVG,3); % Numero degli spindle

frequency = (0.6:0.1:20); % Frequenze di interesse
```

First of all we need to initialize the struct for the all the spectra and also for the spectra we are going to select. We also create a vector for the frequencies of interest.

```
for spindle=1:spindle_count % Per ogni spindle
    data_selection = CTRL_SLOW_SPINDLES_AVG(:,1,spindle); % Singolo trial

    % Calcolo dello spettro
    [pxx, f] = pwelch(data_selection,length(data_selection),[],frequency,fs);

    [peaks, peaks_pos] = findpeaks(pxx); % Individuo i punti di massimo
    max_peak = find(peaks == max(peaks)); % Seleziono il massimo dei massimi
    peak_loc = peaks_pos(max_peak)*(f(2)-f(1)); % Calcolo frequenza del massimo picco

    % Range di frequenze accettabili
    f_min = 8;
    f_max = 12;

    % Se il picco ricade nel range si salva lo spindle (blue) altrimenti si
    % scarta (rosso)
    if peak_loc >= f_min && peak_loc <= f_max
        plot(f,pxx, 'b-');
        ctrl_slow_spectra_select{end + 1} = CTRL_SLOW_SPINDLES(:,:,spindle);
    else
        plot(f, pxx, 'r-')
    end
    grid on
    ctrl_slow_spindle_spectra{end + 1} = pxx;
end
```

We took in consideration the single spindle and we estract the data from the complete matrix. The spectra computation was made trhough the MatLab function pwelch.

`[pxx,f] = pwelch(x,window,noverlap,f,fs)` returns the two-sided Welch PSD estimates at the frequencies specified in the vector, `f`. The vector `f` must contain at least two elements, because otherwise the function interprets it as `nfft`. The frequencies in `f` are in cycles per unit time. The sample rate, `fs`, is the number of samples per unit time. If the unit of time is seconds, then `f` is in cycles/sec (Hz).

In order to find the maximum points of the spectra we used the function findpeaks. After that we found the absolute maximum and the relative frequence, if the frequence of the maximum peak is within the range [8-12] Hz (accepted frequencies) we save the spindle data in the complete struct and also in the selection struct and we plot it in blue, contrariwise if the frequence isn't contained in the accepeted range we save the spindle data just in the complete struct and we plot it in red.

```
%Calcolo delle matrici 3D
CTRL_SLOW_SPINDLE_SPECTRA = cat(3,ctrl_slow_spindle_spectra{:});
CTRL_SLOW_SPINDLE_SELECT = cat(3, ctrl_slow_spectra_select{:});
```

Finally, we used the function cat to create the 3D matrices for all the spectra computed and for the selected spectra.

7) Selected spectra averaging

In this step we proceded to compute the average of the selected spindles (trials), considering all the channels.
The dimensions of the resulting matrices are Spindle length * Number of channels .
For instance, the code below shows the trials averaging of the selected CTRL slow spindles spectra.

```
%Inizializzo la matrice
CTRL_SLOW_SPINDLE_MEAN = zeros(size(CTRL_SLOW_SPINDLE_SELECT,1),size(CTRL_SLOW_SPINDLE_SELECT,2));

spindle_count = size(CTRL_SLOW_SPINDLE_SELECT,3); % Numero di trial selezionati

%Calcolo la media
for spindle=1:spindle_count
    CTRL_SLOW_SPINDLE_MEAN(:,:) = CTRL_SLOW_SPINDLE_MEAN + CTRL_SLOW_SPINDLE_SELECT(:,:,spindle);
end
CTRL_SLOW_SPINDLE_MEAN = CTRL_SLOW_SPINDLE_MEAN ./ spindle_count;
```

For the CTRL fast spindles and for the ICU slow and fast spindles the process is the same.

8) Time averaging

Next we used the results obtained in the previous step and we averaged them over time.

```
CTRL_SLOW_EEG = zeros(1,size(CTRL_SLOW_SPINDLE_MEAN,2));

t_count = size(CTRL_SLOW_SPINDLE_MEAN,1);
for t=1:t_count
    CTRL_SLOW_EEG = CTRL_SLOW_EEG + abs(CTRL_SLOW_SPINDLE_MEAN(t,:));
end
CTRL_SLOW_EEG = CTRL_SLOW_EEG ./ t_count;
```

The method is the same used in the previous averaging. For every row of the matrix we computed the absolute value before calculating the mean over time, in order to highlight the average signal amplitude.
Te code above is related to the CRTL slow spindles, the method is analog for the other matrices.

Finally, we opened eeglab and, after the loading of the channel locations file, we visualized the topographies of the data.

9) Activity localization with Brainstorm

The first thing to do is to save the data obtained in step 7 in four .mat files (one for each matrix). Now we can open brainstorm, create a new protocol and create the subjects (CTRL and ICU).

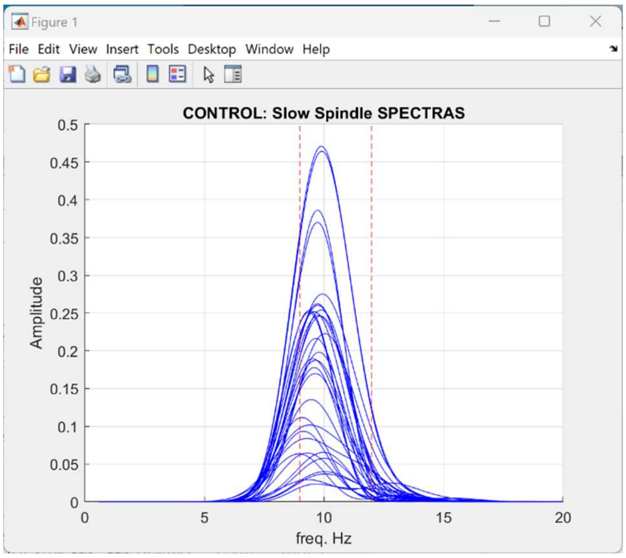
For each subject we have to upload the .mat files conteining the matrices, for the slow and the fast tracks, of the selected spectra averaged over the channels. Then it's necessary to upload the channel positions file and, in association with the MRI registration, project the electrodes on the surface of the scalp.

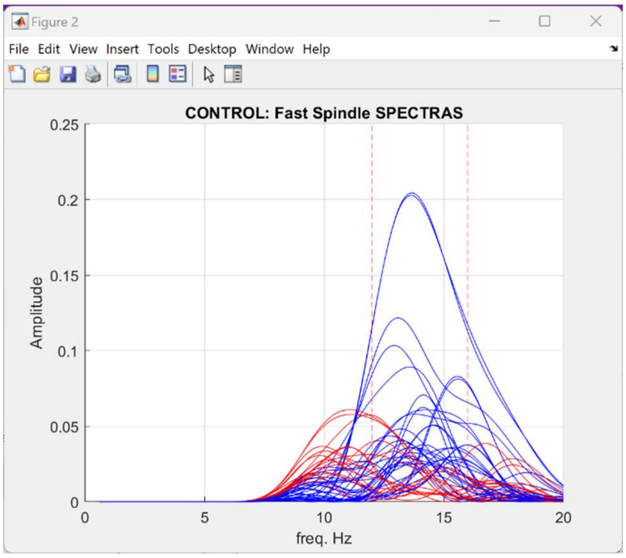
Now we can use the brainstorm functions to compute the head model, the covariance and finally the sources using the sLORETA measure option. This process have to be followed for all the four matrices.

RESULTS AND DISCUSSIONS

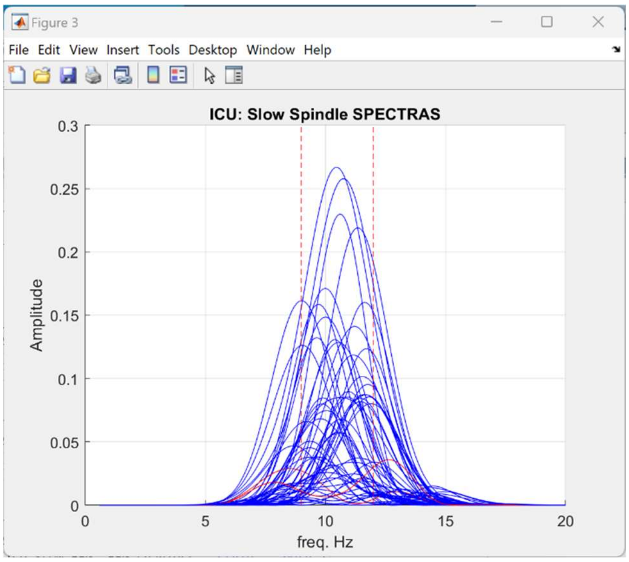
Here are reported the figures and the plots obtained during the analysis.

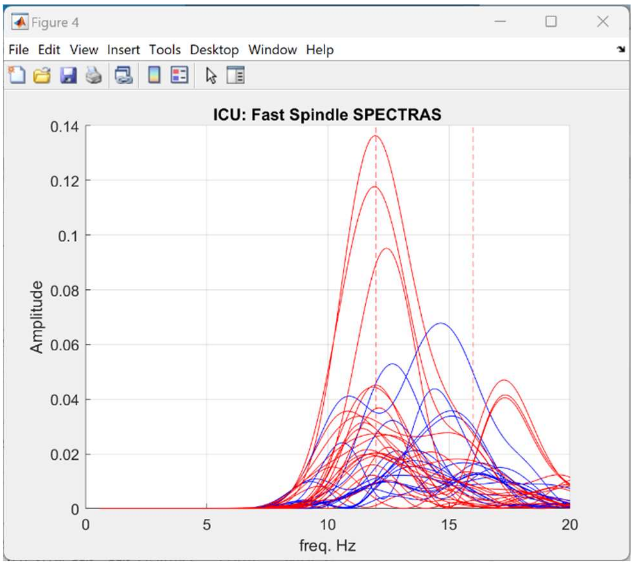
In the spectra plots we set the blue color for the selected spindles spectra and the red color for the exclueded spindles spectra.





Plots of the spectra for the CTRL subject.

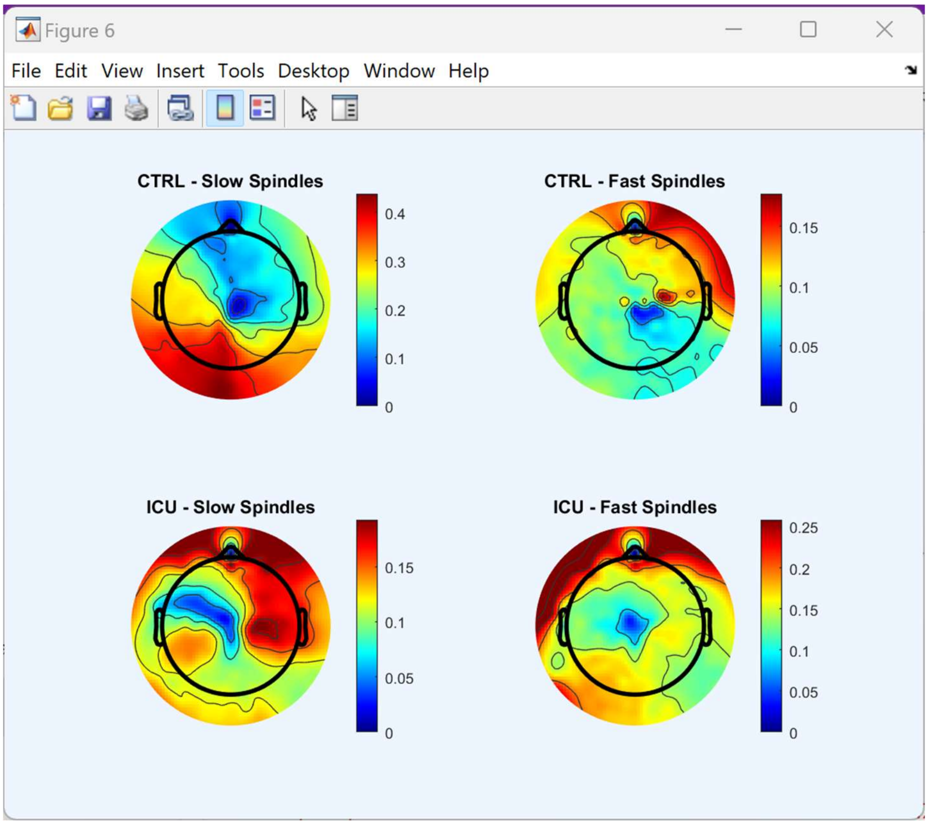




Plots of the spectra for the ICU subject.

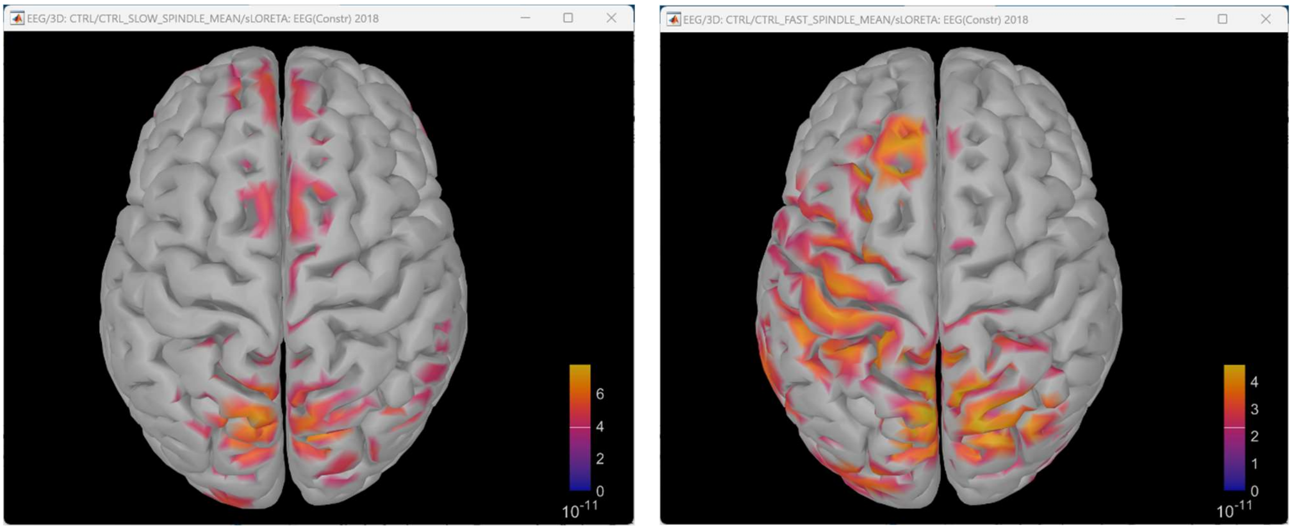
We can notice that almost all the slow spindles spectra are whitin the limits of the frequencies of interest, instead a lot of spectra for the fast spindles are out of the range of interest. We can also notice that the amplitude of the slow spindles spectra is higher respect the fast spindles, from this last observation we can say that slow spindles have higher power, this means that the EEG signal of both the subjects contains slow spindles in larger extent.

Topographies (via eeglab) of the DSP averaged over time for both subject and for the slow and fast spindles:

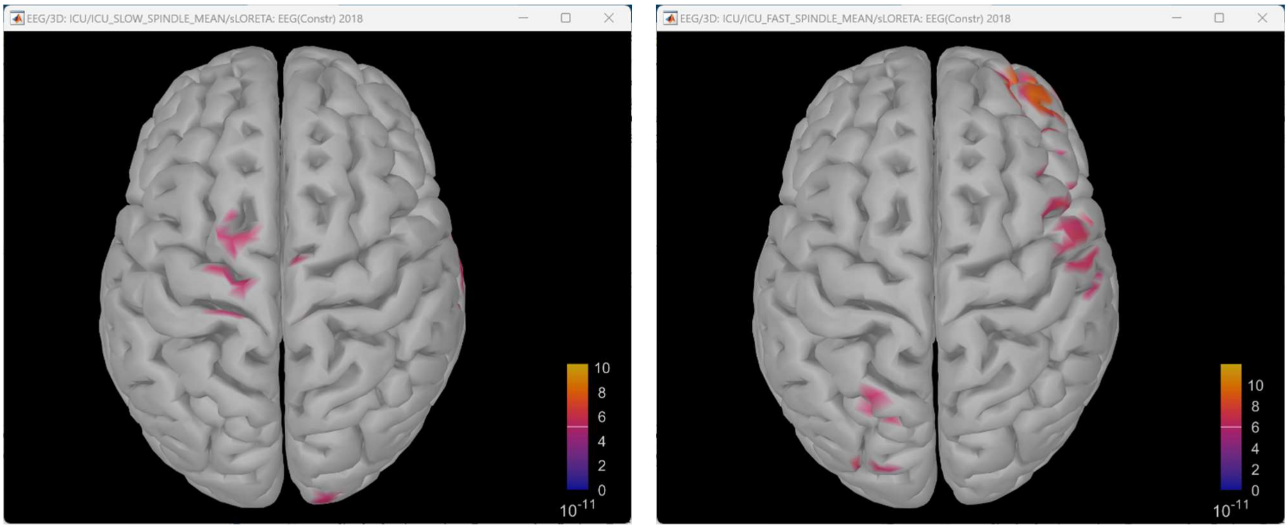


From the topography we can notice that slow spindles shift to more anterior and temporal regions in ICU and they also reduce their power, the fast spindles shift slightly to the anterior region.

Activity localization obtained from the data of step 7 and computed with brainstorm:



Plot of the activity localization for slow and fast spindles in CRTL subject.



Plot of the activity localization for slow and fast spindles for ICU subject.

The activity localization changes for both slow and fast spindles from CTRL to ICU subject, these neurophysiological sleep signatures let us assume that post traumatic stress desease present in COVID-19 survivors could lead to cognitive ipairment.