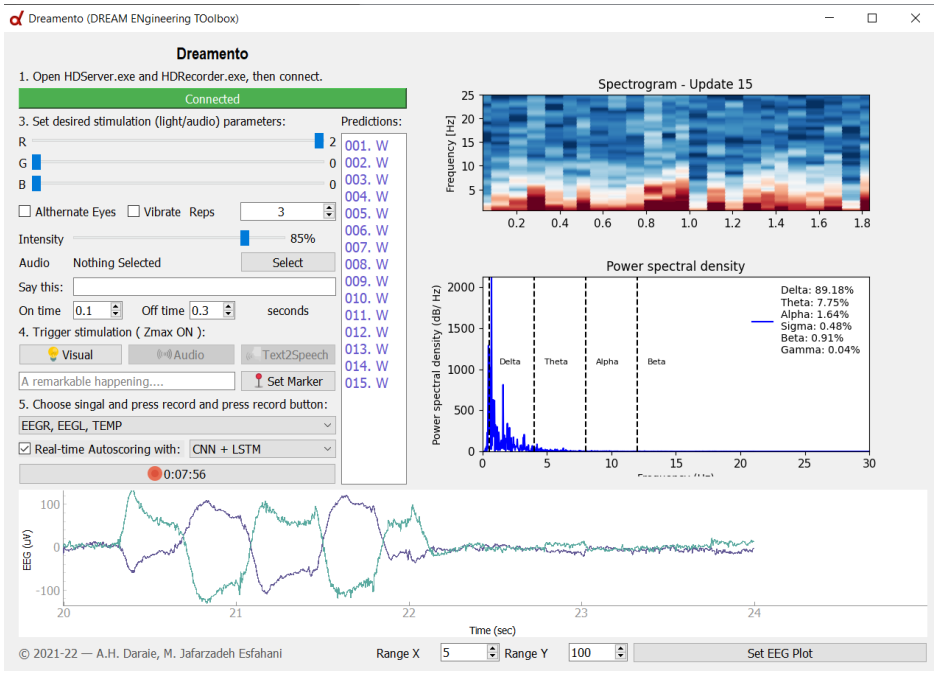
Dreamento user manual

Table of Contents

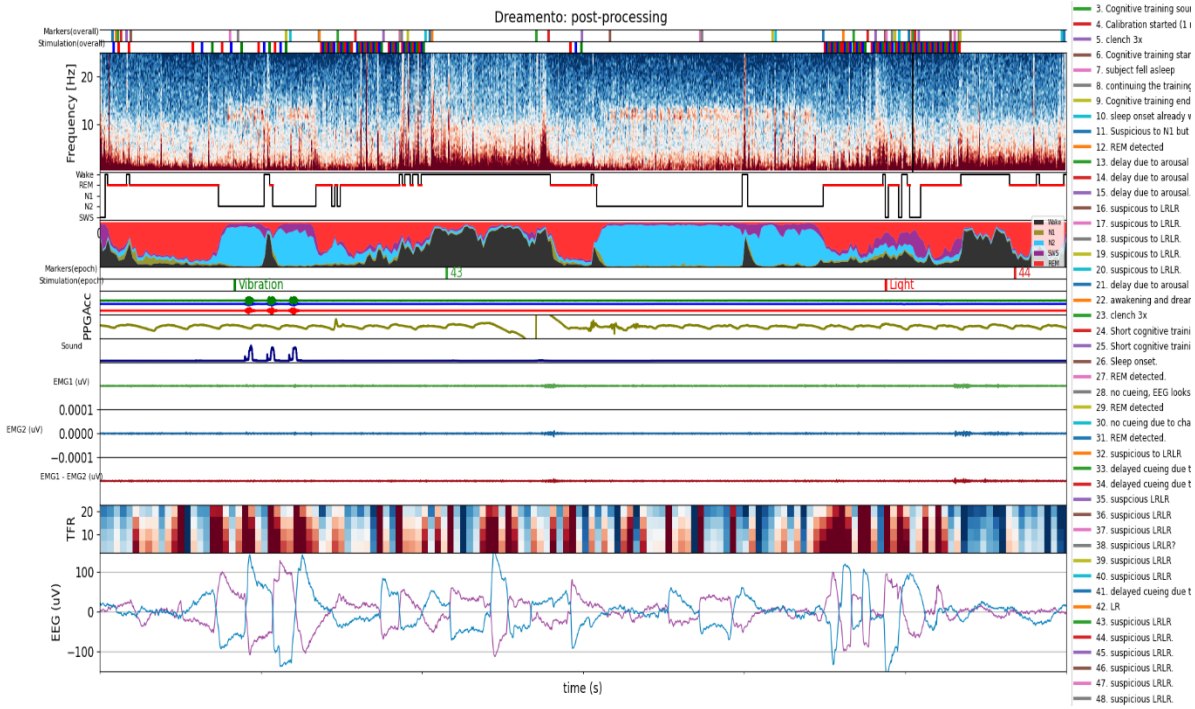
Introduction

Dreamento (DREAM ENgineering TOolbox) is an open-source Python package for:

(1) **Real-time** recording, monitoring, analyzing, and modulating sleep data



(2) **Offline** post-processing of the acquired data, both in a graphical user interface (GUI).



Therefore, the initial factor to take into account is whether you need real-time, offline, or both Dreamento, as the installation and usage procedures for each would vary.

Which Dreamento do you need?

If you need to monitor (and record) **real-time** representation of data along with enabling sensory stimulation, **real-time Dreamento** is the appropriate choice.

If your intention is to **post-process** the acquired data (the data that has been collected either using real-time Dreamento or collected on ZMax sd card), you will specifically require **offline Dreamento**.

If both, simply you need both!

Installation (first-time use only)

Please note that the installation of Offline and real-time is different, and thus two different virtual environments are required as follows. So, install the one you need:

1. ***Real-time Dreamento*:** (A complete video tutorial on how to install real-time Dreamento can be found HERE.)
    a. Open anaconda prompt.
    b. Change direction into where Dreamento is installed *(there you should see "dreamento.yml" file, otherwise you are on the wrong path)*
    c. Create all the required packages on a virtual environment with the following syntax :
    `conda env create --name dreamento --file dreamento.yml`

2. ***Offline Dreamento*:** (A complete video tutorial on how to install Offline Dreamento is provided HERE.)
    a. Open anaconda prompt.
    b. Change direction into where Dreamento is installed *(there you should see "offlinedreamento.yml" file, otherwise you are on the wrong path)*
    c. Create all the required packages on a virtual environment with the following syntax:
    `conda env create --name offlinedreamento --file offlinedreamento.yml`

N.B.1. If you get any error during this process, make sure your directory is changed to where Dreamento main folder from GitHub is cloned. This folder should contain both `dreamento.yml` *and* `offlinedreamento.yml` *files.*

N.B.2. After installation, whenever you need to use Dreamento you should activate the corresponding environment. How? See the next section!

DreamentoConverter

- If you had a recording through ZMax sd card, first make sure you have converted the recorded ".hyp" files into edf. You can do this either using "HDRecorder.exe" or DreamentoConverter. The latter is recommended when you want to convert several ".hyp" files into edf at once. In DreamentoConverter you should specify the path to ".hyp" files to be converted and the required path destinations in order:

```
# define files to be converted
filenames = ['C:\\path\\test_DreamentoConverter\\hypno1.hyp',
'C:\\path\\test_DreamentoConverter\\hypno2.hyp',
'C:\\path\\test_DreamentoConverter\\hypno3.hyp']

# define path to converted folders
destination_folders = ['C:\\path\\test_DreamentoConverter\\converted\\hypno1\\',
'C:\\path\\test_DreamentoConverter\\converted\\hypno2\\',
'C:\\path\\test_DreamentoConverter\\converted\\hypno3\\']
```

Using real-time and offline Dreamento

Make sure you have already installed the required virtual environment. If not, see the previous section!

The very first thing to do is to "activate" the virtual environment so that you can use the toolbox.

If you want to use real-time Dreamento, do the following:

- When you have real-time Dreamento and Hypndoyne software installed, follow the following steps.
- Connect the USB dongle to your pc.
- Run "HDServer.exe".
- Run "HDRecorder.exe" and click on "connect".
- Open the Anaconda prompt, change the directory to where you installed Dreamento

```
cd directory/to/Dreamento
```

- Activate the virtual environment you made in the previous section

```
conda activate dreamento
```

- Run Dreamento through "mainWindow.py":

```
python mainwindow.py
```

When Dreamento's GUI is started, click on "connect". By clicking on the "record" button, the recording will be started!

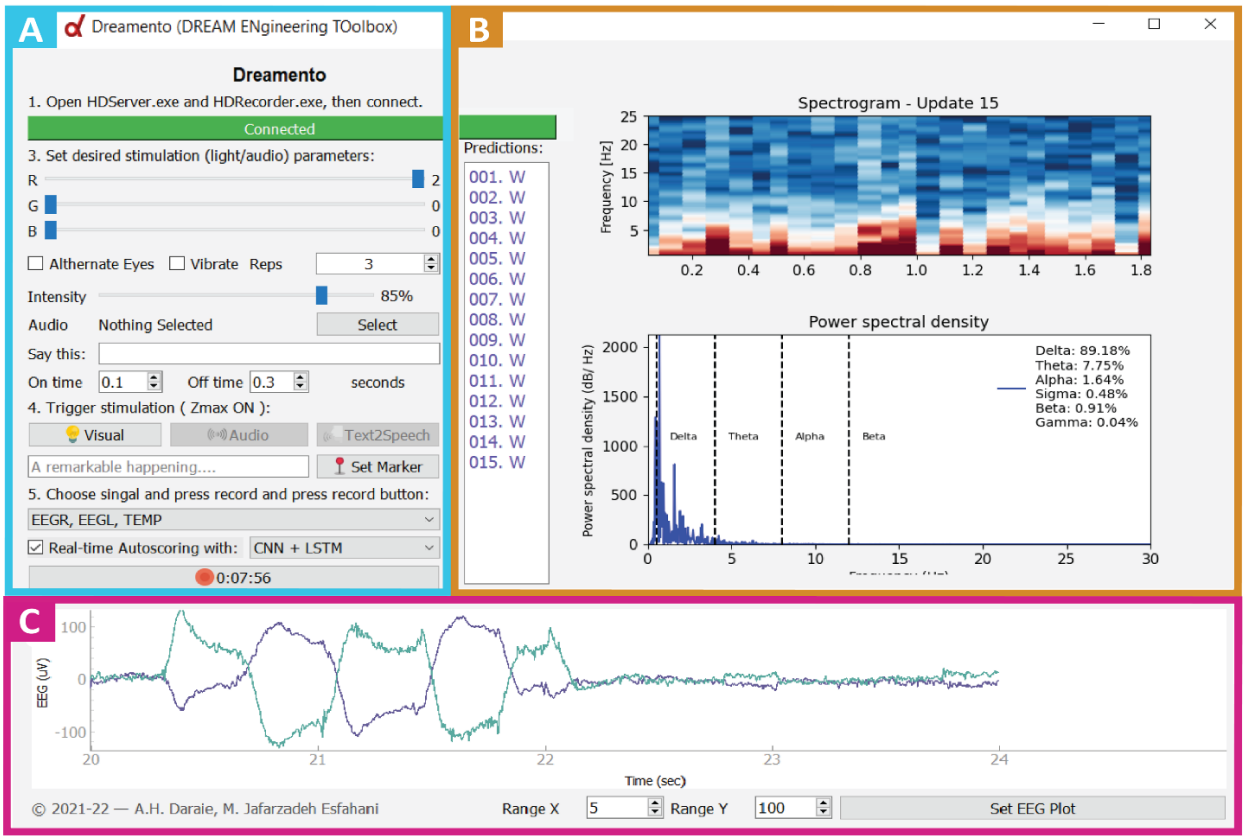
If you want to use **offline Dreamento**, do the following:

- Open Anaconda prompt and activate the virtual environment

```
conda activate offlinedreamento
```

- Then type "spyder" in the anaconda prompt and hit "enter" button.
- In the spyder software, open the "OfflineDreamento.py". If you open it once, it will always stay in spyder as long as you don't close it intentionally. Run the code with the green play button on top and enjoy your data analysis!

Real-time Dreamento Features



In real-time Dreamento you are presented with 3 main sub-panels: **(A) Recording and stimulation panel**, **(B) analysis panel**, **(C) data representation panel**.

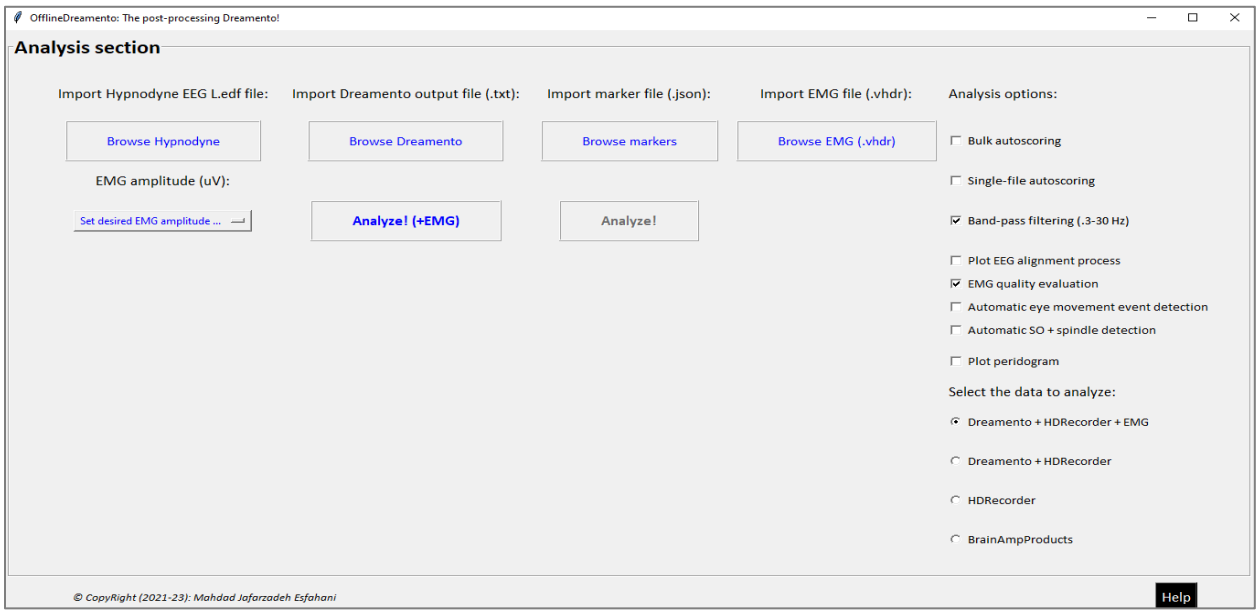
(A) Recording and stimulation panel: in this panel, the user has the features to record the desired set of signals, apply sensory stimulation, and set markers. To start the recording, first, Dreamento should be connected to the server. This can be done by clicking on the "connect" button on top of the panel (this requires HDRecorder and HDServer to be already running). Then by hitting the "record" button at the bottom, data collection would start. Dreamento has the features to apply visual, auditory (either through a pre-recorded audio or using a text-to-speech feature), and tactile stimulation. The features of the stimuli (e.g., the color of light, stimulus on/off duration, number of repetitions, etc) can be specified in this panel.

(B) analysis panel: in this panel, at the end of each 30-second epoch, the data will be analyzed. This will be done through automatic sleep scoring of the last 30 s, creating a time-frequency representation (spectrogram), and plotting the power spectral density. The Spectrogram is capable of keeping results from the last ~ 2 mins for the experimenter to have a better overview of the recent sleep stage transitions.

(C) data representation panel: the real-time data stream is presented here. The x- and y-axis limits can be modified at any time.

Offline Dreamento Features

Four types of data can be analyzed through offline Dreamento: **(1)** ZMax data collected on SD card, **(2)** ZMax data collected in real-time through HDRecorder + real-time Dreamento, **(3)** ZMax data collected in real-time through HDRecorder + real-time Dreamento + another measurement modality such as BrainProducts, **(4)** BrainProducts data (see "select the data to analyze" section in the figure below),



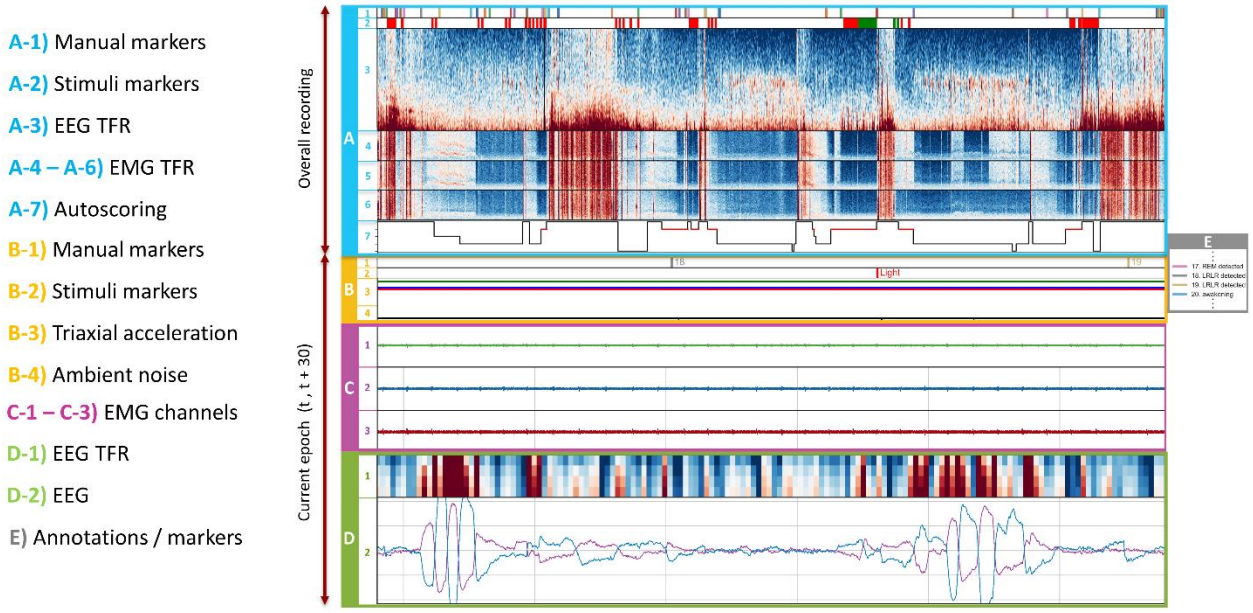
- We cover "bulk autoscoring" in the next sections.
- After selecting the data type to analyze, you can activate different features from the "analysis options". Activating "single-file autoscoring" would result in autoscoring of the results. For the ZMax data autoscoring DreamentoScorer model will be used, whereas YASA will autoscore the BrainProducts data.
- Using "band-pass filtering (.3-30 Hz) is recommended in the majority of the cases.
- "Plot EEG alignment process" is only applicable if you have a simultaneous HDRecorder and Dreamento recording so that you can ensure the automatic alignment was succeeded.
- "EMG quality evaluation" is only applicable if you have a data of type (3): ZMax data collected in real-time through HDRecorder + real-time Dreamento + another measurement modality such as BrainProducts. In this case the spectrograms of the EMG channels will be added to the final plot which helps with the EMG quality evaluation throughout the recording.
- "Automatic eye movement detection" and "automatic SO + spindle detection" will activate automatic algorithms to detect these events throughout the data. Once these checkboxes are activated, the user is presented with a new window from which various detection parameters can be specified. If "single-file autoscoring" is activated, the "Automatic eye movement detection" will only be applied to the epochs determined as

REM sleep and "automatic SO + spindle detection" will be applied to the epochs determined as N1, N2, N3 sleep, otherwise, event detections will be applied throughout the data.

- "Plot periodogram" creates the periodogram of the EEG data.
- Depending on the type of the data you want to analyze, some features may get activated/deactivated.

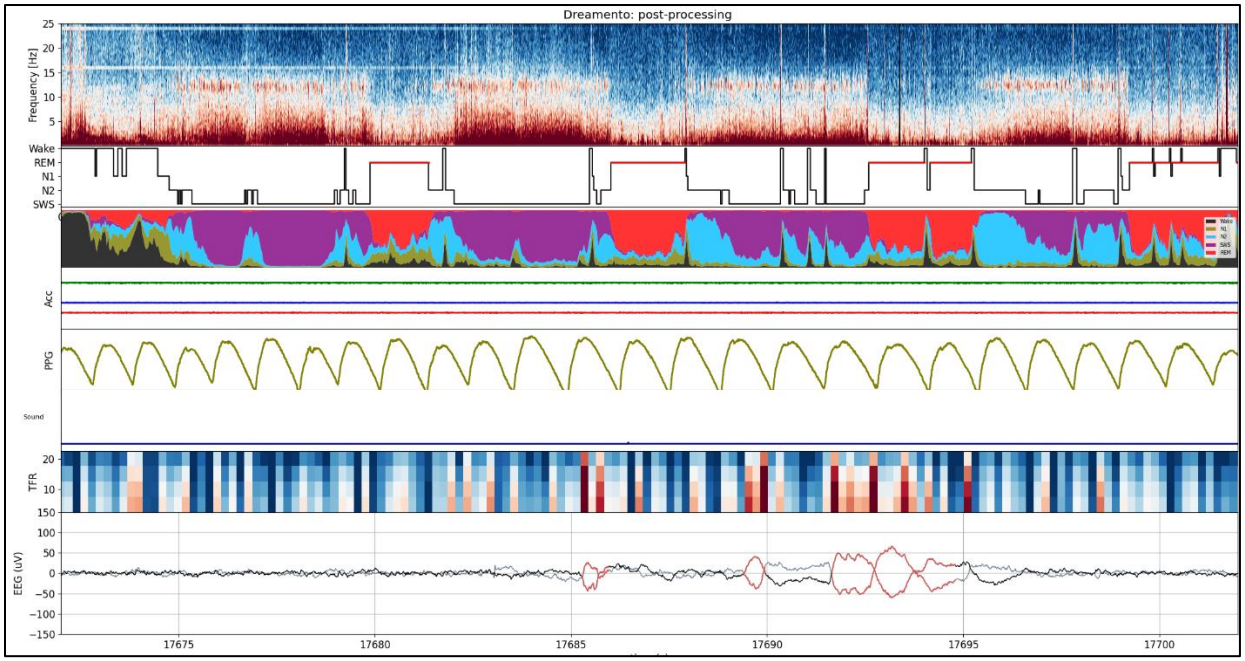
Offline Dreamento results

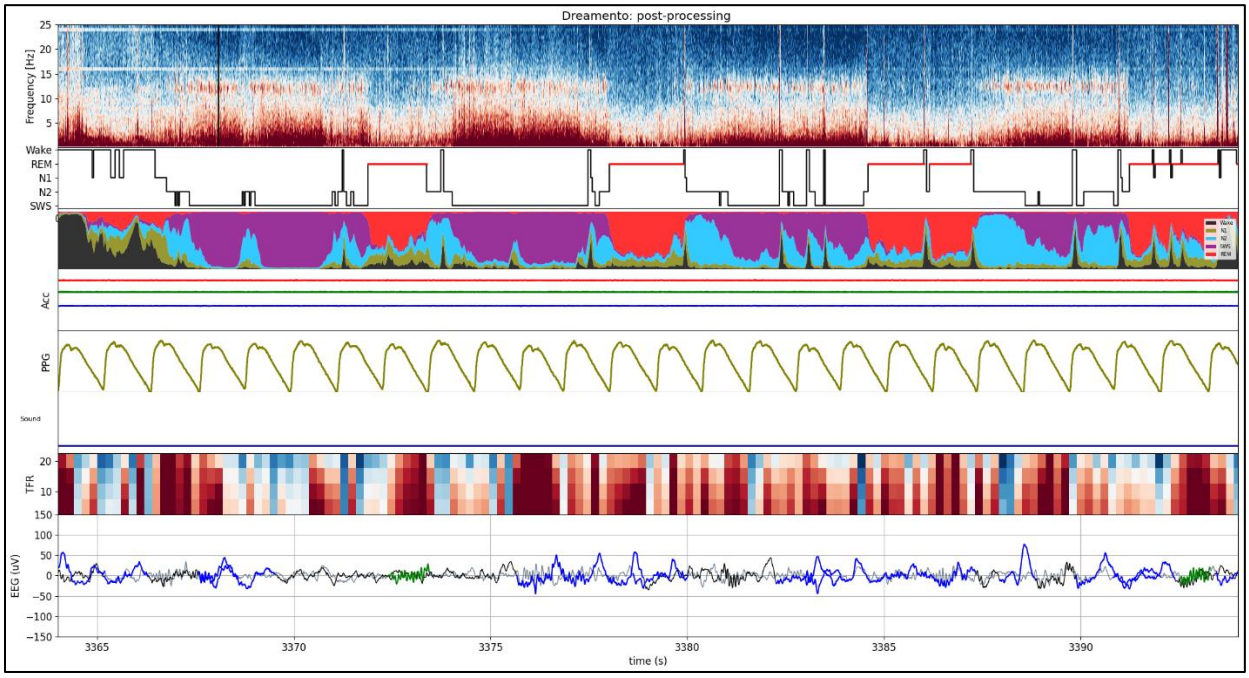
Here we present an example of analyzing a data of type (3) ZMax data collected in real-time through HDRecorder + real-time Dreamento + another measurement modality such as BrainProducts



Automatic sleep event detection

The microstructural features of non-REM sleep such as spindles and slow-oscillaitons (SO) could be automatically detected in ZMax thanks to the algorithm developed in YASA. If the user intends to, the results will be also stored in the same location as the loaded data. Below you can see the example of event detection + the resulting output (REM events are detected in red, spindles in green, and SOs in blue).





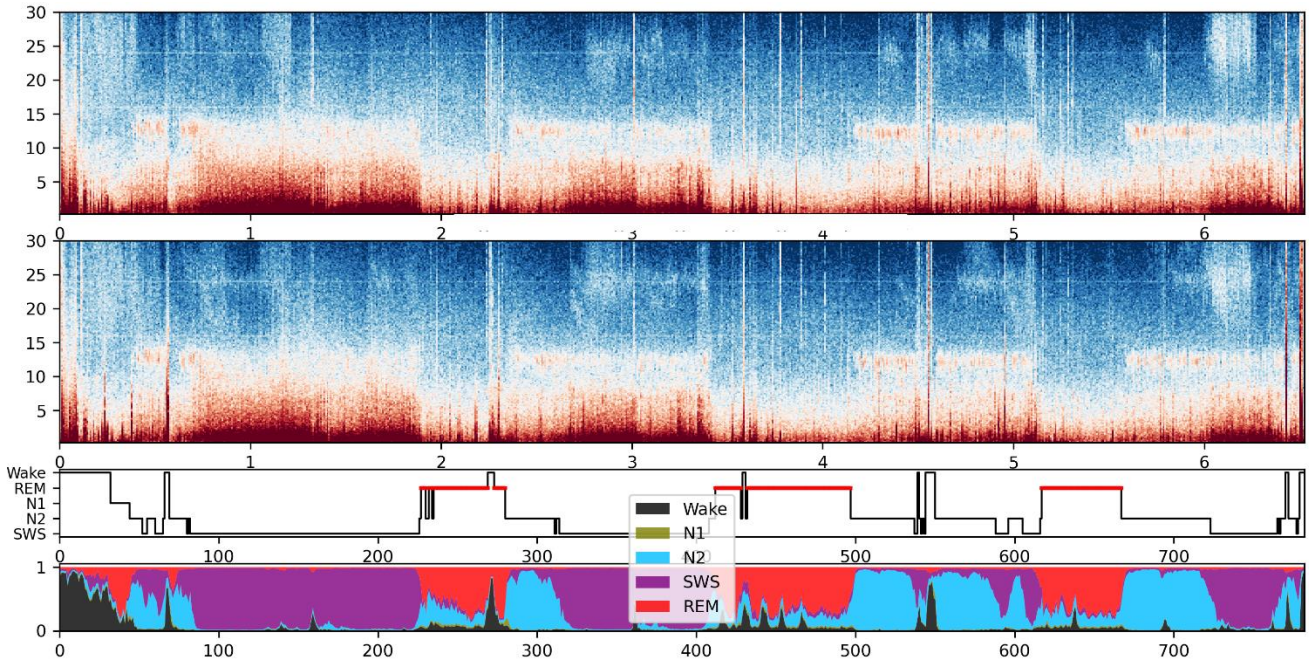
Bulk autoscoring

If you intend to autoscore several ZMax data at once, you may activate the “bulk autoscoring” feature. To provide Dreamento with the files that it needs to autoscore, you need to create a .txt file including the paths to the folders that need to be autoscored. These folders should include all the .edf files that either HDRecorder or DreamentoConverter has created. An example will be a .txt file including these:

```
Path/to/folder/ZMax1/
Path/to/folder/ZMax2/
Path/to/folder/ZMax3/
```

You can also activate whether you want the resulting sleep stages and figures to be saved or not. If you choose to save these, they will be stored in the same path as the original files, `e.g: Path/to/folder/ZMax1/Dreamento_TFR_autoscoring.png (and DreamentoScorer.txt).`

The resulting figures would look like the following, where the spectrogram of both ZMax EEG channels in addition to the predicted sleep stages and the probability of autoscotring for each epoch is demonstrated.



Once the bulk autoscoring process is complete, you will find an extra .xlsx file containing the sleep metrics for all the autoscored data in the same location as the initial .txt file that contained the paths.

Watch us

- Tech for dreaming: link
- Dream x Engineering seminars 2023: link

Citation

For any use, please cite:

Esfahani, M. J., Daraie, A. H., Weber, F. D., & Dresler, M. (2022). Dreamento: an open-source dream engineering toolbox for sleep EEG wearables. arXiv e-prints, arXiv-2207. https://doi.org/10.48550/arXiv.2207.03977