

Traitement EEG

Created jeudi 10 décembre 2015

BrainVision Analyzer 2

Initialisation

Le plus simple de traiter tous les sujets dans un même workspace. Analyzer n'aime pas les sous dossier donc tout les signaux doivent être placer dans un même dossier.

Placer tous les fichiers (en-tête, signaux et marqueurs), de toutes les séquences pour tous les sujets, dans le même dossier.

Commande linux

```
mv /path/to/your/subject*/run*/* /path/to/analyzer/raw
```

Mise en forme des marqueurs de gradient IRM

Si vous traiter des données ASL, il faut modifier les marqueurs de gradients générés par l'IRM pour différencier les gradients *control* des gradient *tag*.

La fonction formatvmrk.m fait ça automatiquement pour les marqueurs de gradient nommés R128. Ils sont renommés en grad_ctr et grad_tag.

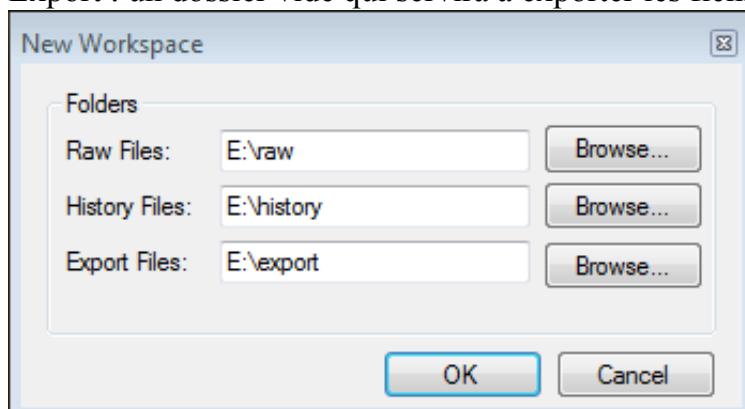
Créer un workspace

Finalement on peut créer un workspace dans Analyzer en indiquant :

Raw : le dossier avec toutes les fichiers

History : un dossier vide qui contiendra l'historique des traitements ainsi que des données en cache (ce dossier pourra être très volumineux)

Export : un dossier vide qui servira à exporter les fichiers traités (évidemment volumineux)



Retrait des "dummy scans"

La première étape consiste à retirer les dummy scans qui sont effectués au début de la séquence. C'est scans produisent des artefacts un peu différents des gradients normaux. Ils ne doivent donc pas être intégrer au débruitage. Le mieux est de les exclure en segmentant le signal d'intérêt.

Placer des marqueurs (Type comment) Start et End au début et à la fin de la période d'intérêt.



Next New Marker

Description:
 Start ▼

Type:
 Comment ▼

Point Width:
 1

☐ Set Channel

All ▼

La segmentation :

Transformations > Segment Analysis Functions > Segmentation

Segmentation Wizard - Step 1

What kind of Segmentation would you like to be performed?

☐ Create new Segments based on a marker position.
☐ Divide data set in equal sized segments.
☐ Set new segments manually.
☒ Create new segments limited by start and end markers.

How should the data be stored?

☒ Do not store data, calculate data on demand.
☐ Cache data to a temporary file.
☐ Cache data to a permanent file.
☒ Cache Data on Request
☐ Store data compressed in history file.

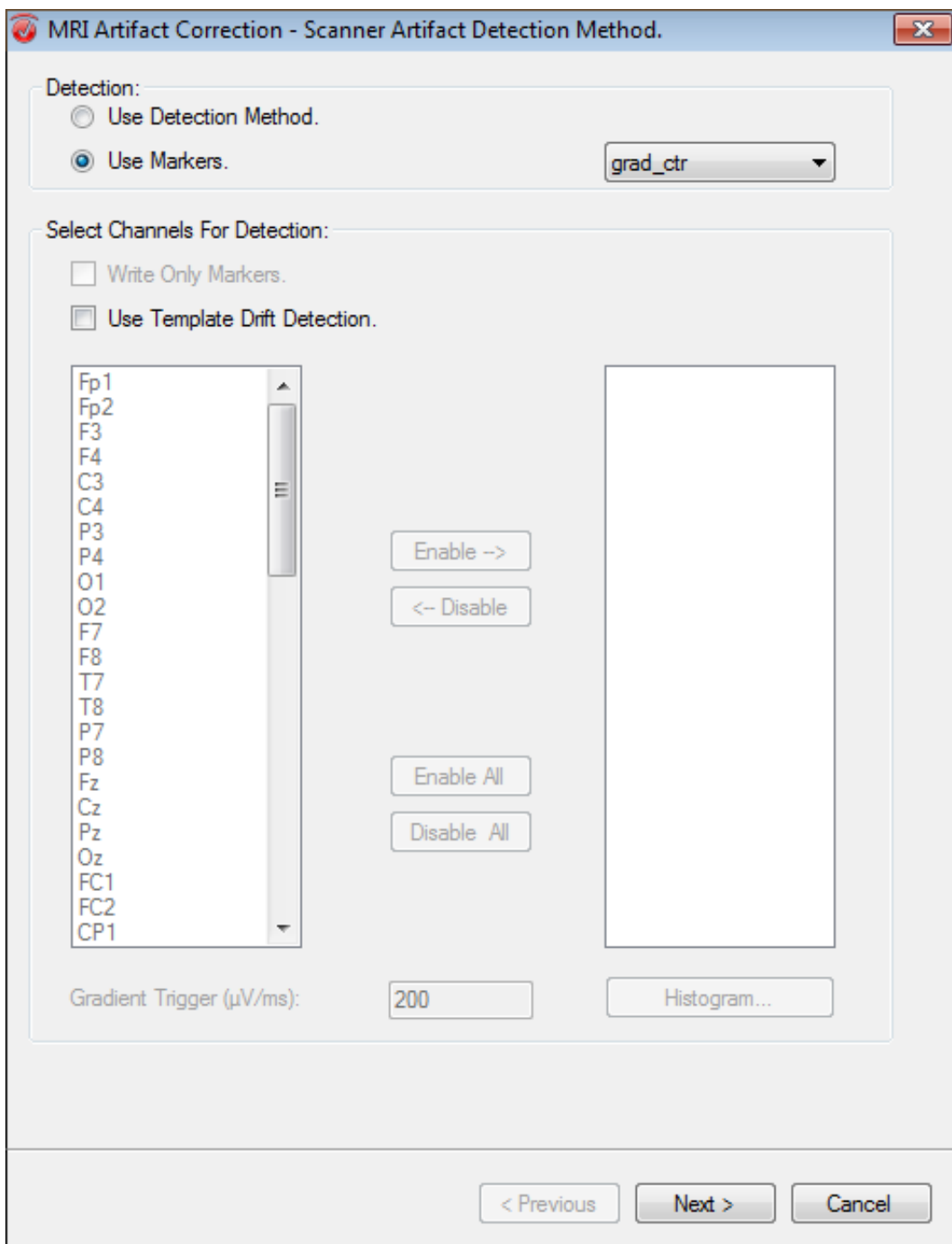
Resolution [nV]: 100

On choisi ensuite les marqueurs Start et End.

Artefacts IRM

Transformations > Special Signal Processing > MR Correction

Méthode : average artefact suppression (AAS)



Utiliser la fonction MR Correction avec comme paramètres :

- Détection à partir du marqueur : grad_ctr (nom choisi pour les gradient *control*)
- Artefact type : Continuous

Based on Time :

- Offset: -100 ms (car l'artefact "bave" un peu avant le marqueur)
- TR: 5520 ms (en ASL on prend deux TRs, dans cet exemple on avait un TR de 2760 ms)

Cocher les options de Baseline

Choisir la méthode Sliding Average Calculation avec un nombre d'intervalles de 31

Cocher la case Common Use of All Channels for Bad Intervals and Correlation

Choisir de corriger toutes les électrodes.

Filtrage et rééchantillonnage

Une fois le template moyen supprimé (méthode AAS), les artefacts de marquage sont complètement supprimés. En revanche il peut rester une partie des artefact de gradient dans le signal.

Pour les nettoyer on peut filtrer le signal :

- coupe-bande pour retirer la fréquence des coupes (~18 Hz dans notre cas)
- passe-bas pour supprimer les harmoniques (on coupe à 30 Hz, la première harmonique étant à ~36 Hz)

Finalement on peut souséchantillonner le signal (disons à 256 Hz), car :

- les autres artefacts, ballistocardiogramme et clignement, sont plus basse fréquence (< 20 Hz)
- on a déjà filtrer passe-bas (à 30 Hz)

Sauvegarde du signal résultant

Étant donné qu'on a énormément réduit la taille du signal, on peut suvegarder le résultat, au moins en cache, pour les futurs traitements.

Artefacts de ballistocardiogramme

Transformations > Special Signal Processing > CB Correction

Détection des battements cardiaques

Transformations > Special Signal Processing > CB Correction

D'après l'écran de configuration, la détection est basée sur deux critères :

- la corrélation avec un événement typique (template)
- l'amplitude du signal en valeur absolue

On choisit une détection semi-automatique pour pouvoir choisir le template et éventuellement changer les paramètres si la détection est mauvaise.

Sélectionner Write Only Markers pour faire uniquement la détection et marquer les pic R dans les marqueurs, sans modifier le signal.

☐ Use Markers
 R-Peak Markers:

☒ Use Peak Detection
 ☒ Semiautomatic Mode

Search Pulse Template:

☐ Use Template Markers

Start [s]:
 Length [s]:

☒ Mark Found Template
 ECG (EEG) Channel:

Pulse Rate:

Pulse Rate [ms]: +/-

Pulse Rate [bpm]: Min: Max:

Correlation and Amplitude:

Correlation Trigger Level:

Amplitude Trigger Level: Min: Max:

Mark Pulses with R-Peak Markers:

☒ Write Only Markers

Après exécution, une première passe de détection est effectuée automatiquement. On peut valider chaque marqueurs R manuellement, ou bien, changer le template et faire une seconde passe. !!! Il est important de bien choisir le template. Il faut prendre un battement propre et ample.

Suppression

Transformations > Special Signal Processing > CB Correction

On utilise la même fonction mais cette fois en se servant des marqueurs R précédemment détectés.

☒ Use Markers
 R-Peak Markers:

Time Delay:

☒ Use Whole Data to Compute the Time Delay

☒ Use Following Interval to Compute the Time Delay:

Start [s]: Length [s]:

Use Following Value for Time Delay [s]:

Total Number of Pulse Intervals Used for Average:

Correct Following Channels:

ECG
EMG1
EMG2

Enable -->
 <-- Disable

Enable All
 Disable All

Fp1
Fp2
F3
F4
C3
C4
P3
P4
O1
O2
F7
F8
T7
T8
P7
P8
Fz
Cz
Pz

Si on est limité par la place, il faut mieux ne pas stocker les données en cache.

Export des données traitées

Export > Generic Data

Cocher Write header file et Write marker file, Test format (.vhdr, .vmrk).

Cocher ensuite :

Data file format: Binary format

Data orientation: Multiplexed

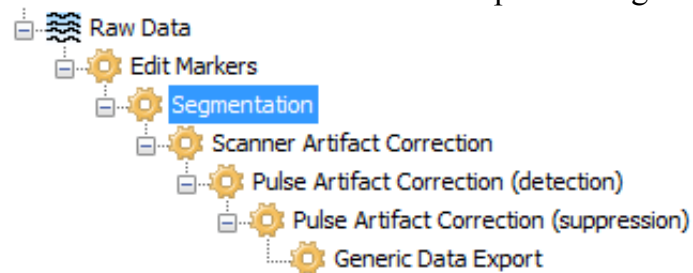
Line Delimiters: UNIX format

IEEE 32 bit floating point format

Sélectionner toutes les électrodes.

Reproduction des traitements

La chaîne de traitement est terminer pour un signal.



Il ne reste plus qu'à les reproduire sur tous les signaux.

Pour chaque nouveau signal, il faut d'abord faire manuellement l'étape Edit Markers. Puis, on glisse l'étape Segmentation (du premier signal) sur l'étape Edit Markers du nouveau signal. Une intervention est nécessaire pour valider la détection, si on a opté pour une détection semi-automatique.

Fieldtrip

Cette partie fait référence à des codes Matlab développés spécifiquement pour un protocole. Ils faudraient sûrement les adapter à d'autres données.

Initialisation

Installation de Fieldtrip

Télécharger la toolbox Fieldtrip pour Matlab
Ajouter le dossier principal de Fieldtrip dans le path Matlab.
exécuter

```
ft_defaults
```

Import des signaux

Placer les données dans un dossier export, un sous-dossier par sujet, et un sous-dossier par session (run)
/path/to/data/export/<subject>/<run>

Artefacts oculaires

Cette méthode est basée sur une décomposition en composante indépendante (ICA) de toutes les voies EEG, afin d'extraire la composante comportant les artefacts de clignements d'yeux (blinks). Cette composante est détectée en évaluant la corrélation de chaque composante avec la voie Fp1. La plus forte valeur de corrélation détermine la composante de blink.

Exécuter

```
SCRIPT_fieldtrip  
Qui initialise la toolbox fieldtrip et appelle ensuite  
script_reject_blink
```

Brainstorm

Initialisation

Mise en forme des marqueurs de stim

Création du protocole

Anatomies

Signaux EEG