

# TP3 - IMN530

FOUQUET, Jérémie et MÉTHOT, Vincent

28 avril 2014

## Question 1 IRM fonctionnelle

Plusieurs outils d'analyse existent déjà pour traiter des données d'IRMf. Nous avons choisis de les utiliser directement plutôt que de les implémenter en python. Deux suites logicielles ont retenu notre intérêt (puisque nous les connaissions déjà), soit FSL [<http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/>] et AFNI [<http://afni.nimh.nih.gov/>], qu'il faudra avoir installé pour faire fonctionner le script associé à ce numéro (`Q1_IRMf.sh`).

### a Étapes de reconstruction

Il faut garder à l'esprit qu'à chaque étape de reconstruction, il est fortement conseillé d'inspecter visuellement les données. Dès leur réception, on a visuellement inspecté plusieurs tranches de *fmri.nii* à tous les temps pour s'assurer que les artéfacts n'étaient pas trop importants et que la correction de mouvement n'était pas nécessaire (voir Fig. 1), comme mentionné dans la question. De plus, nous avons effectué une transformée de Fourier des séries temporelles (voir Fig. 2).

Les étapes de reconstruction suivantes furent appliquées sur les données d'IRMf. Le résultat de chacune est illustrée sur la Fig. 3. On y voit sur la première ligne l'image originale (3A - *fmri.nii*) et les séries temporelles de quelques voxels voisins (3B). Sur la deuxième ligne, il y a le résultat de la première étape de reconstruction (3A & B *fmri\_smooth.nii.gz*), c'est-à-dire l'image débruitée. Cette étape a pour but d'augmenter le rapport signal sur bruit par une convolution avec une gaussienne de FWHM = 6mm. La troisième ligne montre le résultat du filtrage temporel sur chaque voxel (3E *fmri\_low.nii.gz*). On voit par le fait même que la valeur moyenne de chaque voxel est fixée à 0, ne laissant que les variations temporelles. Comme du bruit et des artéfacts de nature physiologique se retrouvent aux hautes fréquences, nous avons décidé de couper les fréquences supérieures à 0.06 Hz (sachant que notre signal idéal à une fréquence fondamentale de 0.02 Hz) de chacune de nos séries temporelles (3F). La dernière étape de reconstruction consiste à calculer voxel par voxel une corrélation entre le signal idéal (contenu dans le fichier *Data/ideal.txt*) et mesuré. On en tire une carte de coefficients de corrélation entre -1 et 1 (3G). On pourra ensuite seuiller cette carte pour obtenir un masque binaire des zones d'activations. La comparaison entre 3B et 3H montre l'effet du processus de reconstruction sur les données d'IRMf autant au niveau spatial que temporel.

### b Segmentation

On a effectué la segmentation des zones d'activation à l'aide de l'outil graphique d'AFNI, de manière à trouver les paramètres idéaux en fonction des zones que l'on s'attendait à voir activées. Puisque la tâche Roland est une tâche requérant des fonctions motrices, de mémoire et de vision, on s'attend à voir

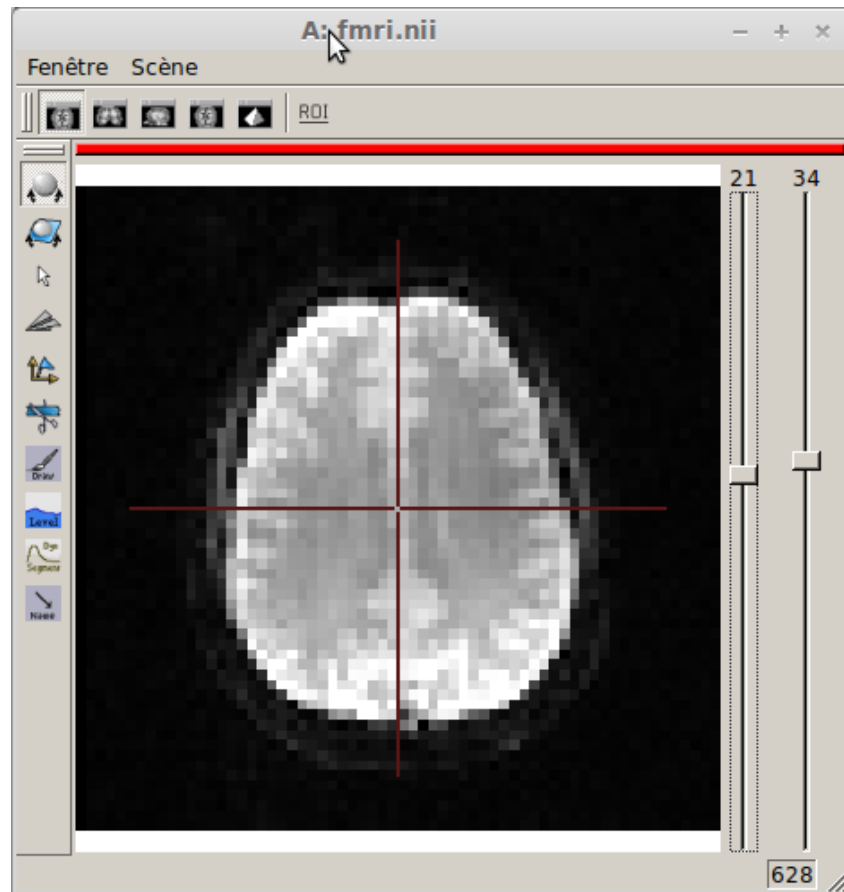


FIGURE 1 – Inspection visuelle de fmri.nii dans anatomist. On peut inspecter plusieurs tranches pour tous les points temporels à l’aide des deux curseurs à droite, comme dans un film.

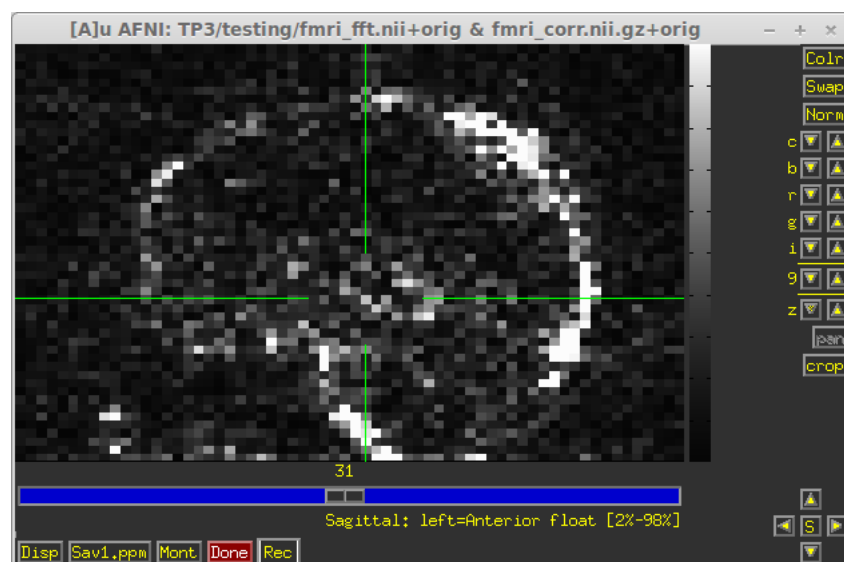


FIGURE 2 – Coupe sagittale de la transformée de Fourier de fmri.nii à chaque point. Une fréquence près de 0.02 Hz (la fréquence fondamentale du stimulus) est affichée dans AFNI. Les zones blanches peuvent être dues à de l’activation cérébrale mais aussi (et surtout) à un effet de bord relié à du mouvement.

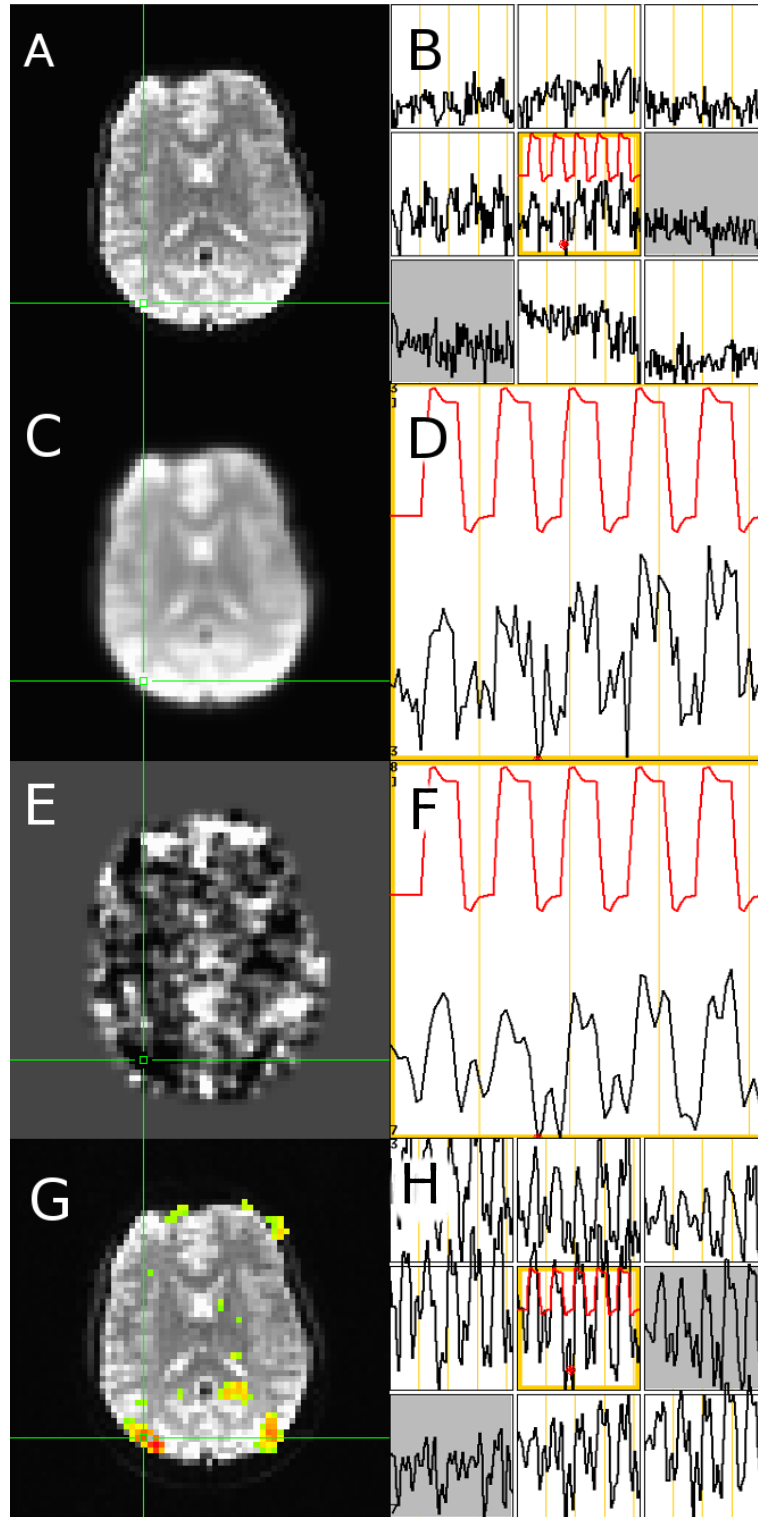


FIGURE 3 – A - Image originelle. B - Séries temporelles pour quelques voxels adjacents de l'image originelle. C - Image débruitée spatialement (convolution avec une gaussienne de FWHM = 6mm). D - Série temporelle débruitée. E - Image après passage d'une filtre passe-bas temporel. La valeur moyenne (fréquence nulle) représentant l'information structurale est perdue. F - Série temporelle de l'image filtrée. G - Corrélation point à point (seuillé à  $\rho > 0.5$ ). H - Séries temporelles pour quelques voxels adjacents de l'image reconstruite. Position du point sélectionné :  $x = 22, y = 13, z = 24$ . Les séries temporelles rouges représentent le signal idéal. Celles en noir représentent le signal mesuré/transformé.

« allumer » 6 zones reliées à ces activités. Les tâches motrices sont principalement localisées dans le cortex moteur (gyrus précentral). La mémoire est principalement localisée dans l'hippocampe. La vision est principalement localisée dans le cortex occipital. Nous nous sommes donc servis de nos connaissances anatomiques *a priori* pour réaliser la segmentation et le regroupement des voxels activés (*clustering*).

L'étape de segmentation dans le script `Q1\_IRMf.sh` réalise l'opération de seuillage et de regroupement automatiquement. Elle donne sept régions distinctes, chacune d'entre elles étant étiquetée avec une valeur différente :

1. Lobe occipital droit (vision)
2. Lobe occipital gauche (vision)
3. Hippocampe gauche (mémoire)
4. Hippocampe droit (mémoire)
5. Gyrus précentral droite (mouvement)
6. Région hors du cerveau (artéfact)
7. Gyrus précentral gauche (mouvement)

La méthode de segmentation utilisée consiste en un seuillage simple des valeurs de coefficient de corrélation (on ne garde que celles entre 0.5149 et 1), puis le *clustering* de AFNI enlève les voxels n'étant pas regroupés. Plus précisément, les groupes comportant moins de 50 voxels voisins sont mis à zéro.

## c Zones d'activation

## Question 2 IRM de diffusion

### a Estimation des tenseurs

La fonction `Q2_IRMd.tenseur` utilise la méthode de la pseudo-inverse pour effectuer le calcul des tenseurs. Elle peut prendre en entrée un masque qui indique pour quels voxels calculer les tenseurs. Sont également mis à 0 tous les éléments de tenseurs qui :

1. Correspondent à un signal à  $b = 0$  nul.
2. Prennent une valeur NaN ou Inf.

La fig. 4 illustre les tenseurs que nous avons ainsi obtenus sur une carte de FA (calculée grâce à l'algorithme présenté dans la section suivante).

### b FA et ADC

La fonction `Q2_IRMd.compAdcAndFa` calcule l'ADC et la FA à partir d'un champ de tenseur tel que calculé par la fonction `Q2_IRMd.tenseur`. La fig. 5 illustre pour certaines tranches l'ADC et la FA, respectivement. Les unités de l'ADC sont les mêmes que celles du coefficients de diffusion, soit  $[longueur]^2/[temps]$ , alors que la FA est sans unité.

### c Tractographie

La fonction `Q2_IRMd.tracking` effectue une tractographie déterministe sur un champ de tenseurs. Les paragraphes suivants expliquent quelques enjeux importants de cette fonction.

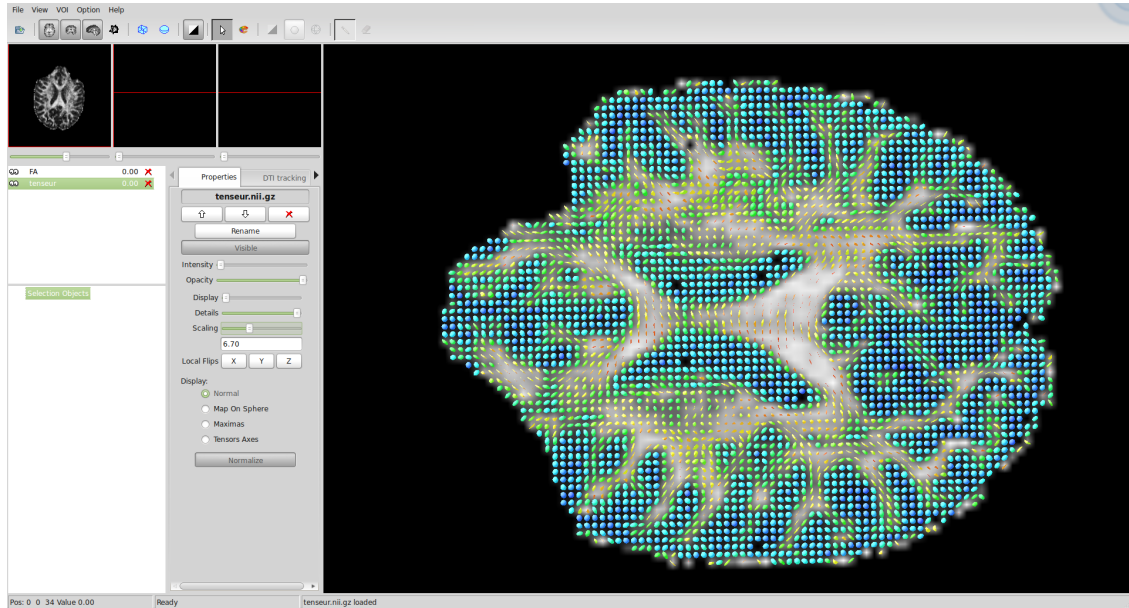


FIGURE 4 – Illustration dans le Fibernavigator des tenseurs que nous avons obtenus grâce à la fonction `Q2_IRMd.tenseur`. Les tenseurs sont superposés à la FA. À noter que seule une région identifiée comme étant le cerveau grâce à l’algorithme BET de FSL est présentée.

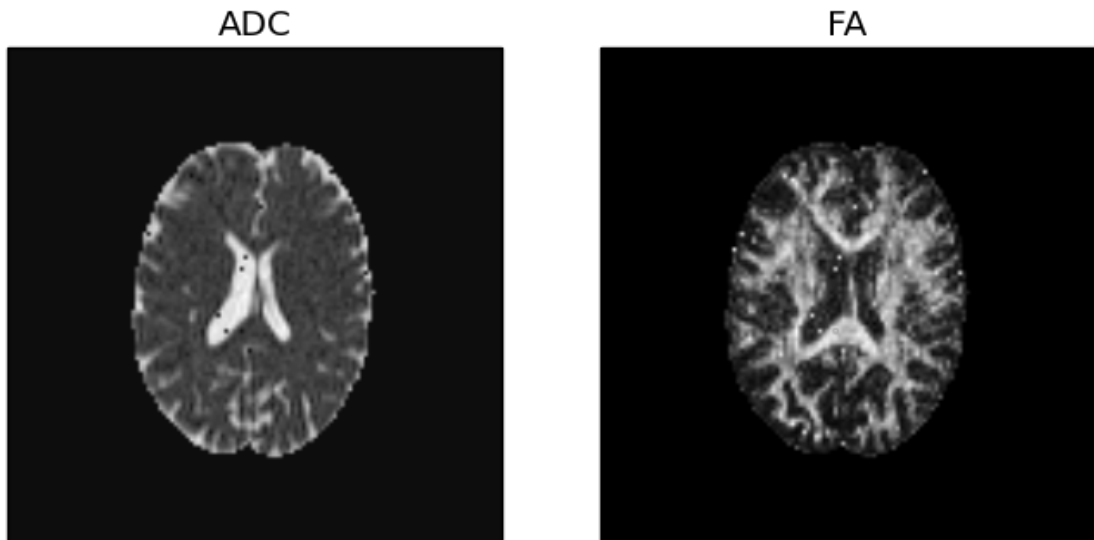


FIGURE 5 – ADC et FA pour une tranche axiale du volume d’IRMd.

| Champ interpolé                         | Nb d'interp. | Nb de calculs | Vitesse de calcul | Précision possible |
|---|--------------|---------------|-------------------|--------------------|
| Mesures d'IRMd                          | 64           | 3             | Basse             | Haute              |
| Tenseurs avant (après)<br>décomposition | 6            | 2 (1)         | Moyenne           | Moyenne            |
| Direction du vecteur principal          | 2            | 0             | Haute             | Basse              |

TAB. 1 – Champs pouvant être interpolés lors d'un pas de tractographie, avec le nombre d'éléments devant être interpolés pour chacun d'entre eux (colonne 2) ainsi que le nombre de calculs manquants avant d'arriver à une direction de propagation (colonne 3). Plus ces nombres sont grands, plus le temps de calcul est grand. Cependant, en descendant dans les rangées du tableau, de l'information est également perdue avant l'interpolation. Ceci pourrait rendre l'interpolation moins précise.

**À quelle étape interpoler ?** Tout d'abord, lors de l'implémentation de la tractographie, nous avons dû prendre une décision concernant la question suivante : *quel champ doit être interpolé à chaque pas de la tractographie ?* Nous avons considéré les options présentées dans le tab. 1. Ces options ont toutes des avantages et des inconvénients : plus le nombre d'éléments à interpoler et le nombre de calculs à faire avant d'arriver à une direction de propagation sont grands, plus le temps de calcul total de l'algorithme sera grand. Cependant, comme on le voit dans le tab. 1, la précision possible du résultat final augmente aussi lorsque ces nombres augmentent. Il en est ainsi car en général, ce qui permet d'interpoler moins d'éléments génère une perte d'information précédant l'interpolation (les transformées effectuées ne sont en effet pas réversibles). Les interpolations faites par la suite risquent donc d'être moins précises. Nous avons choisi une méthode qui semblait être un compromis entre temps de calcul et précision possible, soit **l'interpolation des tenseurs**. Nous n'avons cependant pas eu le temps de tester l'impact réel sur les temps de calcul et la précision des résultats.

**Méthode d'interpolation** La méthode d'interpolation a un impact crucial sur la vitesse de notre algorithme. Nous avons essayé les méthodes : plus proche voisin, linéaire et spline de degré  $n$ , avec  $n$  allant de 2 à 5. Avec une méthode plus complexe que la méthode linéaire, l'interpolation devenait très visiblement le goulot d'étranglement de la vitesse de notre algorithme. Cependant, cela n'a pas causé d'amélioration apparente de la qualité de notre tractographie. Les autres résultats présentés dans ce rapport ont donc été obtenus avec la méthode d'interpolation linéaire.

**Résultats** La figure 6 illustrent des fibres obtenues pour différents pas et différents seuil sur la FA, avec 10000 graines placées aléatoirement.

## Question 3 Fusion

### a Justification

**Espace de recalage** Pour choisir l'espace de référence dans lequel effectuer les recalages, nous avons cherché l'espace de l'image qui détenait

1. La meilleure résolution
2. Le moins d'artéfacts
3. Le plus haut SNR

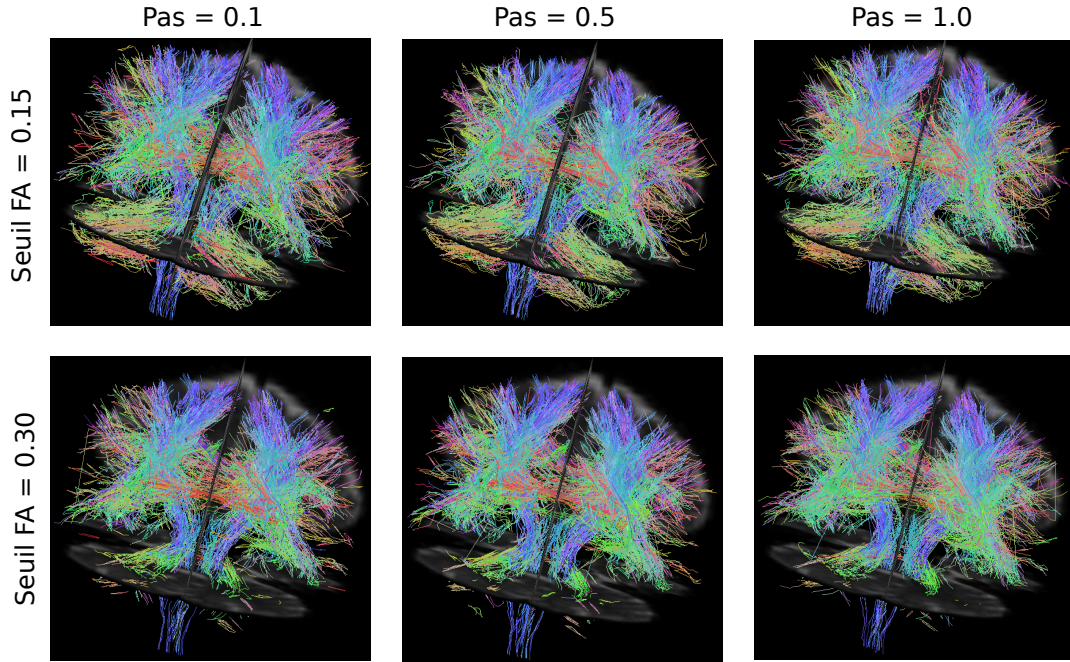


FIGURE 6 – Fibres obtenues avec 10000 graines placées aléatoirement pour différentes valeurs de pas et de seuil sur la FA visualisées dans le Fibernavigator et superposées à l'image à  $b = 0$ . On voit bien une augmentation du nombre et de la longueur des fibres lorsque le seuil diminue. Cela est particulièrement visible dans le cervelet. Le pas ne semble pour sa part pas avoir une influence majeure sur l'apparence des fibres.

Nous voulions en effet recalcr les images sur celle qui représentait le plus fidèlement et le plus précisément la réalité, ce qui était reflété par ces caractéristiques. L'espace choisi a donc évidemment été l'espace de l'image T1.

**Outil** Tout d'abord, mentionnons que pour l'IRMd, nous avons recalé l'image à  $b = 0$  sur l'image T1. C'est en effet elle qui possédait le plus haut SNR et se rapprochait le plus d'une image anatomique habituelle. En IRMf, nous avons recalé la moyenne temporelle du signal sur l'image T1 afin de maximiser le SNR de l'image recalée. De plus, pour les deux volumes recalés, nous avons remarqué que le processus était grandement facilitée si les cerveaux étaient « extraits » avant le recalage. Nous avons donc appliqué la fonction `fs15.0-bet` sur tous les volumes avant le recalage, avec des paramètres permettant une bonne extraction du cerveau.

Ensuite, nous avons choisi le logiciel de recalage ANTS pour la grande variété des possibilités qu'il offre de même que pour sa réputation. Étant donné les différences importantes entre les pondérations des images recalées, les mesures de similarité que nous avons envisagées sont l'information mutuelle (IM) et la corrélation-croisée (CC).

Pour recalcr l'IRMd sur la T1, nous avons d'abord essayé une transformation affine (il est toujours bon d'essayer une méthode plus simple pour commencer). Les résultats étaient déjà relativement prometteurs, mais quelques éléments de l'image ne semblaient pas encore parfaitement recalés, peut-être en raison d'artéfacts (fig. 7). Nous avons donc essayé quelques combinaisons de recalage non-linéaire qu'offrait ANTS. Après inspection visuelle des résultats, nous avons subjectivement arrêté notre choix sur le recalage avec une CC de rayon 4, une transformation de type élastique avec un pas de gradient de 3, une régularisation gaussienne sur le gradient de la transformation de paramètres 0.5 et 3 et des



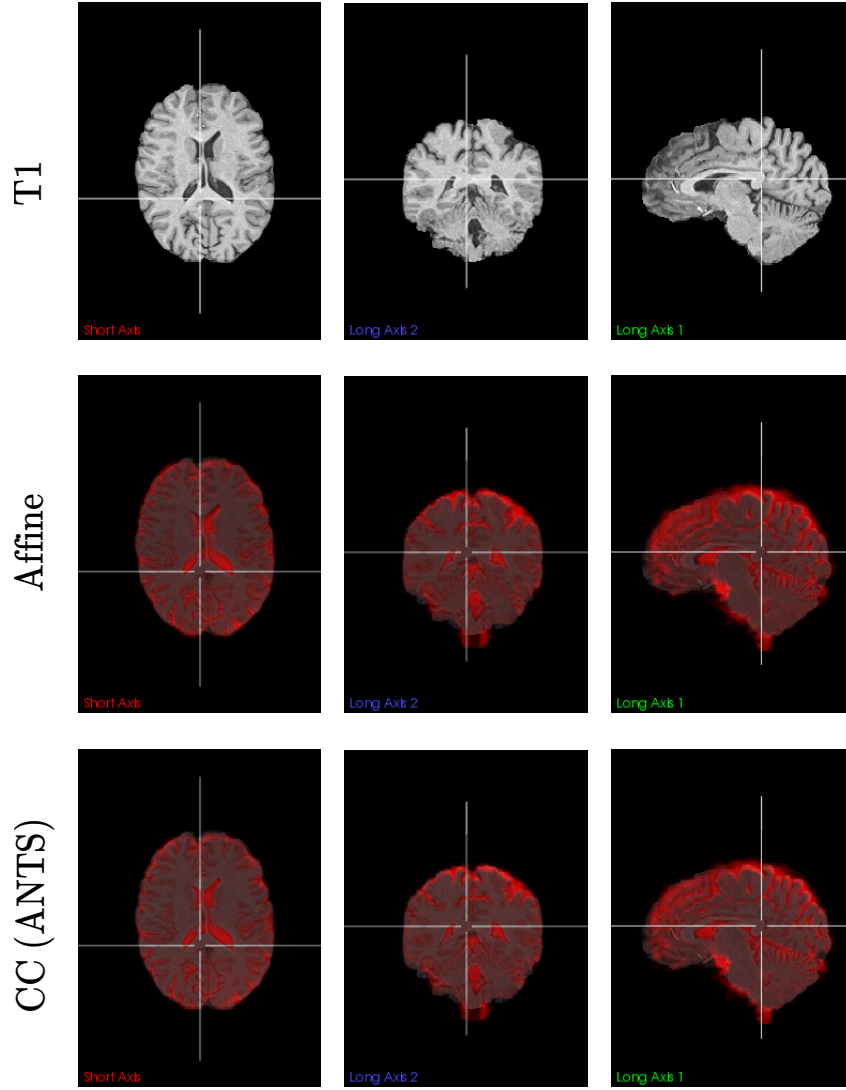


FIGURE 7 – Recalage de l’image d’IRMd à  $b = 0$  (en rouge) sur l’image T1 à l’aide d’une transformation affine (ligne 2) et d’une transformation non linéaire avec ANTS utilisant les paramètres suivants en langage ANTS : `-m CC[... ,1,4] -t Elast[3] -i 30x20x10 -r Gauss[0.5,3]`.

nombre d’itérations de 30, 20 et 10 sur les échelles de résolution  $x/4$ ,  $x/2$  et  $x$ , avec  $x$  la résolution initiale (en langage ANTS, `-m CC[... ,1,4] -t Elast[3] -i 30x20x10 -r Gauss[0.5,3]`). Les résultats sont illustrés sur la fig. 7.

Pour recalculer l’IRMf sur la T1, nous avons utilisé la même approche que pour l’IRMd. L’image d’IRMf présentait cependant une résolution spatiale encore plus faible que l’IRMd, et son SNR semblait aussi assez faible. Il était donc plus difficile de qualifier le recalage. Nous avons finalement arrêté notre choix sur la même méthode de recalage non linéaire que pour l’IRMd. Les résultats sont illustrés sur la fig. 8



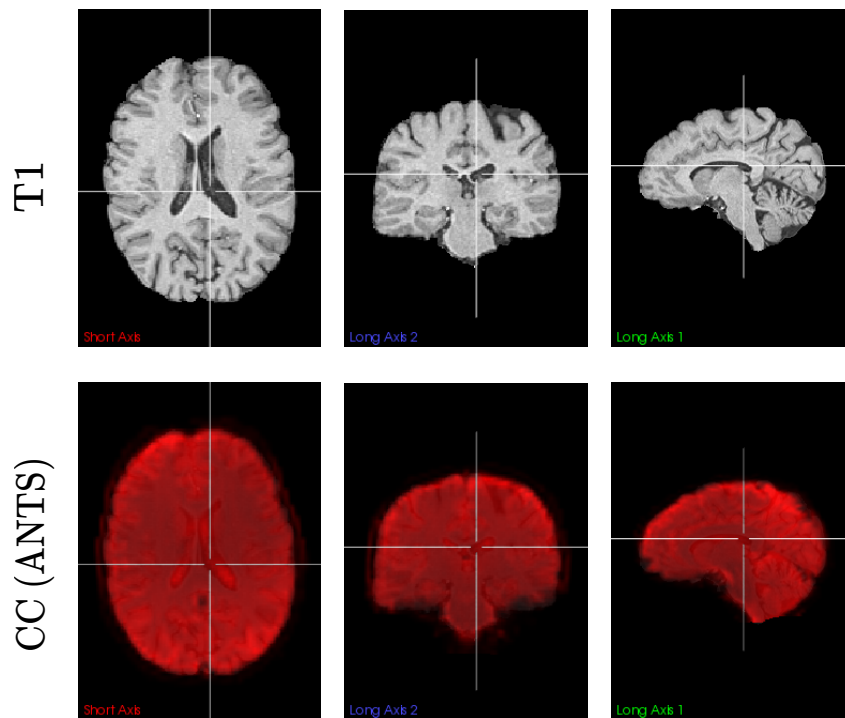


FIGURE 8 – Image T1 (rangée 1) et recalage non linéaire de l'image d'IRMf moyennée dans le temps (en rouge, rangée 2) sur l'image T1 effectué avec ANTS avec les paramètres suivants, en langage ANTS : `-m CC[... ,1,4] -t Elast[3] -i 30x20x10 -r Gauss[0.5,3]`.

## b Connectivité des zones fonctionnelles

### Question 4 Bonus

a FA et ADC

b Tractographie avec Dipy