

Para pasar de un nifti a otros dos que guarden los valores correspondientes a los manojos en dirección paralela y perpendicular a los streamlines usamos el siguiente script de Python:

```
/home/alonso/Dropbox/datos_estudiantes/aylin/Python/deNiftiObtenerParalelayPerpendicular.py
```

Este script toma como parámetros el nombre de la carpeta con los archivos (después de .../ses-P30/dwi/), el nombre de los PDDs, el nombre del archivo que indica el número de clusters y el nombre del archivo nifti del que se van a sacar los valores

Ejemplo

```
python
/home/alonso/Dropbox/datos_estudiantes/aylin/Python/deNiftiObtener
ParalelayPerpendicular.py
out_dPar_SMT_dParMinMaxDTL1_Kaden_07_new/
intra_mus_dPar_SMT_dParMinMaxDTL1_Kaden_Uniform_07.nii
intra_n_clusters_dPar_SMT_dParMinMaxDTL1_Kaden_Uniform_07.nii
intra_uniformity_dPar_SMT_dParMinMaxDTL1_Kaden_Uniform_07.nii
```

Esto va a dar como resultado en la misma carpeta 2 archivos con el nombre del último parámetro más “_direccion_paralela.nii” y “_direccion_perpendicular.nii” para los archivos nifti que corresponden al valor en la dirección de los streamlines y la dirección perpendicular a los streamlines correspondientemente.

Luego, se usa el siguiente script de bash para convertir estos archivos nifti en los archivos de texto que usan los streamlines.

```
/home/alonso/Dropbox/datos_estudiantes/aylin/bash/crear_mapas_streamlines_20_para_1_caracteristica.sh
```

Este script recibe un parámetro; el nombre del archivo nifti. Si el archivo está dentro de más carpetas que “../ses-P30/dwi/” se tiene que añadir.

Ejemplo

```
bash
/home/alonso/Dropbox/datos_estudiantes/aylin/bash/crear_mapas_streamlines_20_para_1_caracteristica.sh
out_dPar_SMT_dParMinMaxDTL1_Kaden_07_new/intra_uniformity_dPar_SMT_dParMinMaxDTL1_Kaden_Uniform_07_direccion_paralela.nii
intra_uniformity_dPar_SMT_dParMinMaxDTL1_Kaden_Uniform_07_direccion_paralela
```

Análisis características

Luego, para tomar en cuenta estos nuevos archivos de texto se deben de añadir a un archivo con la estructura del archivo

```
/home/alonso/NAS/NAS/datos/aylin/nombresCaracteristicas.txt
```

Después de esto, se puede correr el script de R que hará el análisis de estas características de la siguiente manera

```
Rscript analisisCaracteristicas.R nombrePath /path/direcciones.txt /path/out --valorenpc1
--boxplots --simulados
```

Ejemplo

```
Rscript
/home/alonso/Dropbox/datos_estudiantes/aylin/R/analisisCaracterist
```

```
icas.R  
/home/alonso/NAS/NAS/datos/aylin/displaciasCorticales/derivatives/  
/home/alonso/NAS/NAS/datos/aylin/nombresCaracteristicas.txt  
/home/alonso/NAS/NAS/datos/aylin/prueba2 --valorenpc1 --boxplots
```

Análisis permutaciones

Para hacer la prueba de permutaciones se necesitan los mismos parámetros que para análisis de características más un número extra en donde se decide la cantidad de permutaciones que se van a tomar en cuenta. Para este script hay que tomar en cuenta que ya para 100 permutaciones dura unas pocas horas.

```
Rscript analisisPermutaciones.R nombrePath /path/direcciones.txt /path/out 100  
--simulados --generados
```

El flag de --generados es para usar los archivos .Rda anteriormente generados si es que ya se ejecutó este código anteriormente y solo que quieren imprimir otras imágenes que se añadan después para evitar volver a esperar horas.

Ejemplo

```
Rscript  
/home/alonso/Dropbox/datos_estudiantes/aylin/R/analisisPermutaciones.R  
/home/alonso/NAS/NAS/datos/aylin/displaciasCorticales/derivatives/  
/home/alonso/NAS/NAS/datos/aylin/nombresCaracteristicas.txt  
/home/alonso/NAS/NAS/datos/aylin/prueba1 100
```