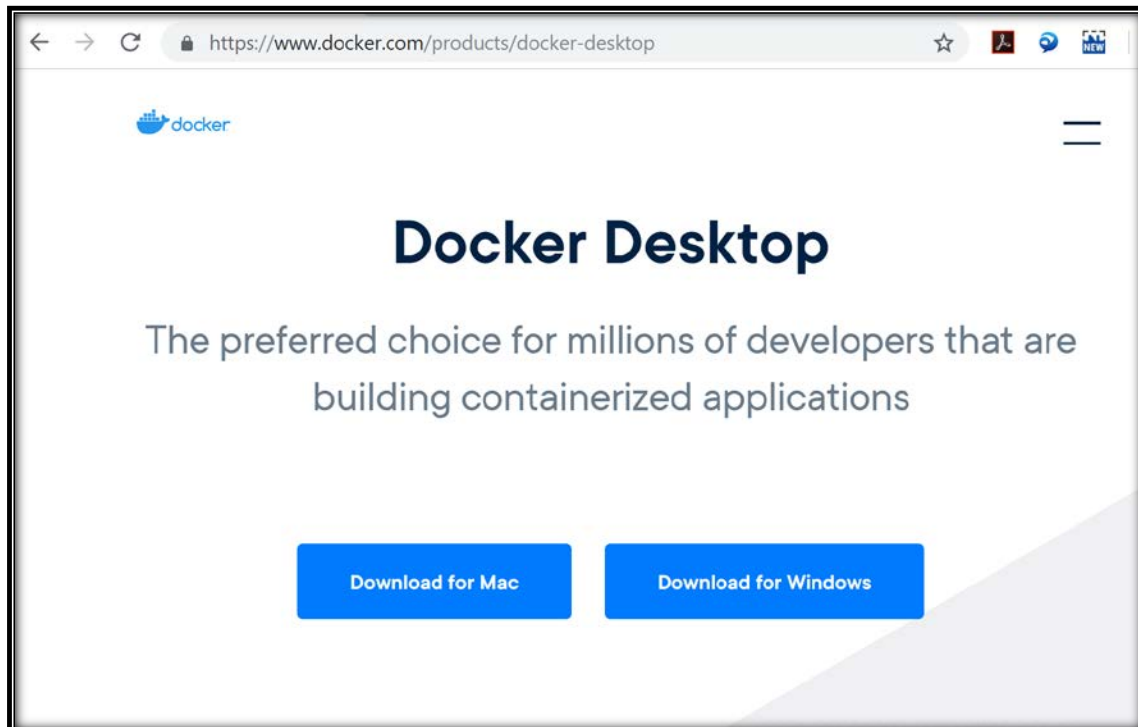




LittleBrain manual

1. To download Docker Desktop, go to: <https://www.docker.com/products/docker-desktop>. Download the Docker version that is compatible with your operating system (e.g. windows, mac). You may need to create a Docker account before downloading Docker.





2. Once the program “Docker” is installed, start the program “Docker” in your computer. Then open the terminal of your computer (mac: look for “Terminal”; windows: look for “Command Prompt”). Type the following in your terminal in order to download the LittleBrain docker container (this will use approximately 2.5GB of space):

If using mac or linux, type this:

```
docker pull xaviergp/littlebrain:3.0.0
```

If using windows, type this:

```
docker.exe pull xaviergp/littlebrain:3.0.0
```

3. To start LittleBrain, follow these steps:

3.1 - Start the program “Docker” in your computer.

3.2 - When Docker is running (it may take a while for Docker to start running), open the terminal of your computer (mac or linux: look for “Terminal”; windows: look for “Command Prompt”). Then type the following in your terminal:

If using mac or linux, type this in the terminal:

```
docker run --rm -p 8888:8888 xaviergp/littlebrain:3.0.0
```

If using windows, type this in the terminal:

```
docker.exe run --rm -p 8888:8888 xaviergp/littlebrain:3.0.0
```

3.3 - Copy the password that has been generated after the word “token” shown in your terminal (shown in red in the image below).

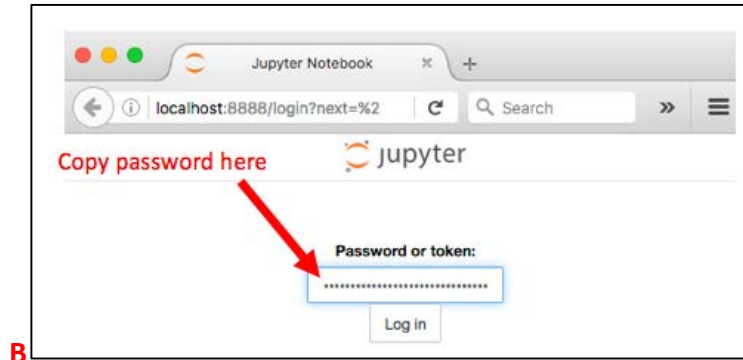
```
INFO: /root/matlab/startup.m does not exist ... creating
Some packages in this Docker container are non-free
If you are considering commercial use of this container, please consult the relevant
license:
https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Licence
[I xx:xx:xx.xxx NotebookApp] Writing notebook server cookie secret to
/root/.local/share/jupyter/runtime/notebook_cookie_secret
[I xx:xx:xx.xxx NotebookApp] Serving notebooks from local directory: /src
[I xx:xx:xx.xxx NotebookApp] 0 active kernels
[I xx:xx:xx.xxx NotebookApp] The Jupyter Notebook is running at:
[I xx:xx:xx.xxx NotebookApp] http://62136cb37e2d:8888/?token=72c3aa5b5d18320d2ff5a2fa2beff5db7040ba7fb48d1370
[I xx:xx:xx.xxx NotebookApp] Use Control-C to stop this server and shut down all
kernels (twice to skip confirmation).
[C xx:xx:xx.xxx NotebookApp]

Copy/paste this URL into your browser when you connect for the first time,
to login with a token:
http://64156cb37e3d:8888/?token=72c3aa5b3d18320d2ff5a2fa2beff5db7040ba7fb48d1370&token=72c3aa5b3d18320d2ff5a2fa2beff5db7040ba7fb48d1370
```

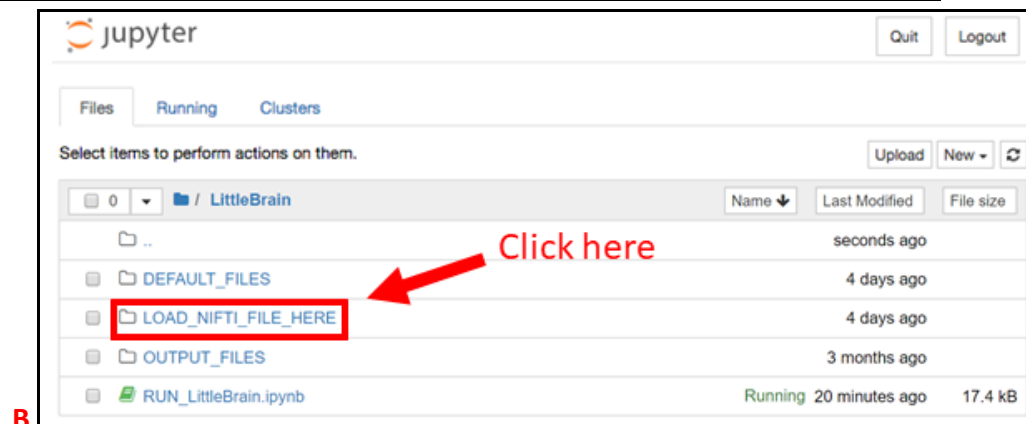
If an error occurs, re-start Docker and repeat step 3 (this will solve the issue if port 8888 is being used by a previously opened jupyter notebook).



3.4 - Open your internet browser (for example, firefox), type the address “localhost:8888”, and copy the password in the text box that will appear:

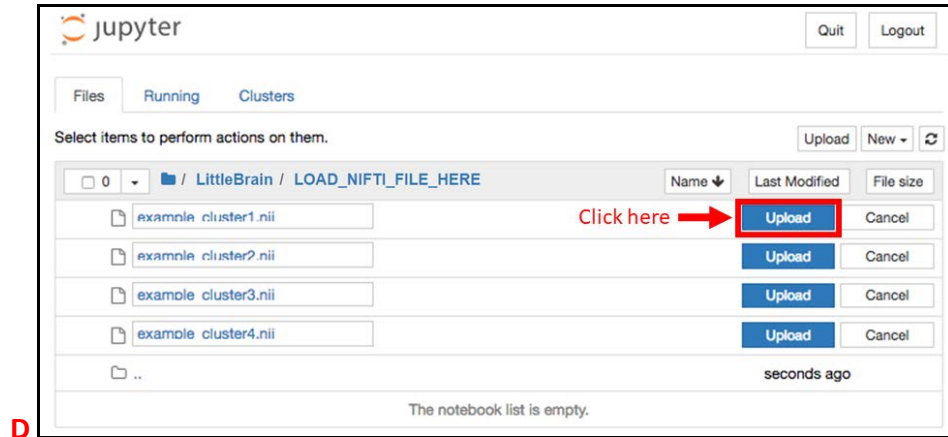
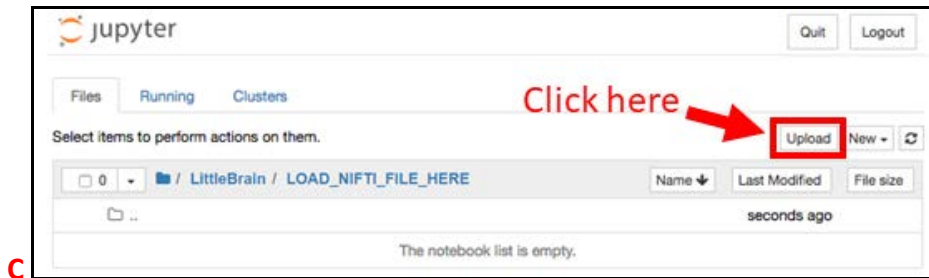


4. To upload the clusters that you want to map, go to “LittleBrain”, then go to “LOAD_NIFTI_FILE_HERE”, and then click “Upload”.

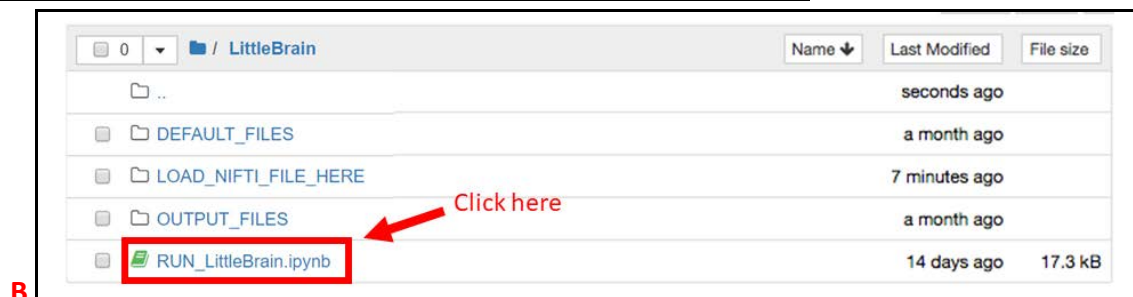
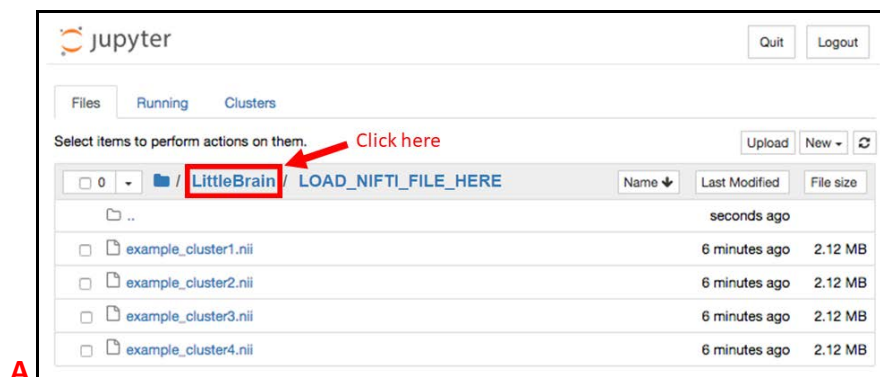




LittleBrain version 3.0.0 manual



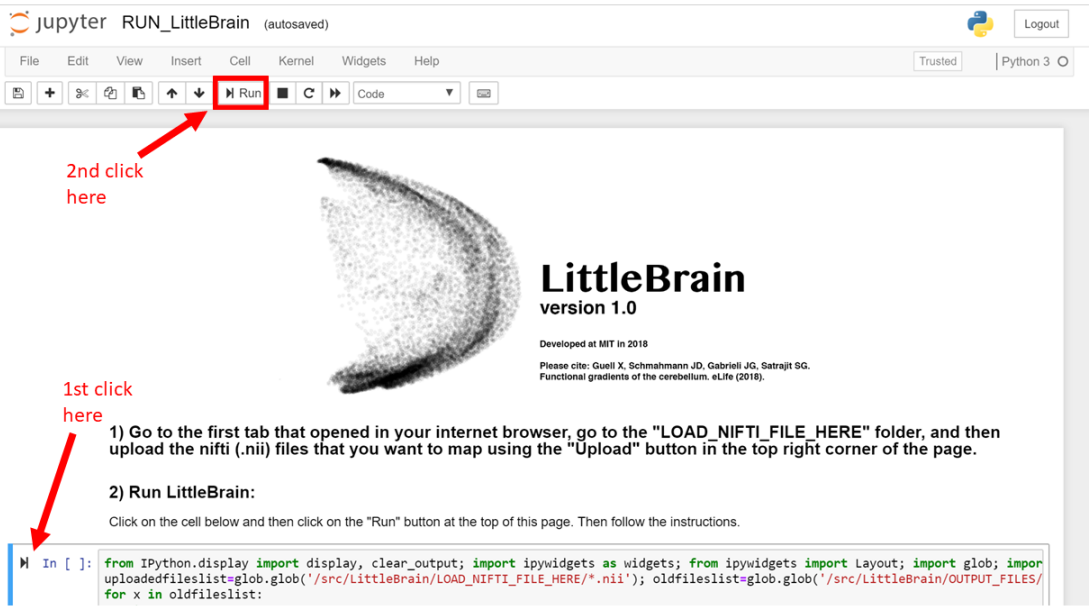
5. To start LittleBrain, go back to the “LittleBrain” folder and click “RUN_LittleBrain”





6. Follow the instructions that will appear in the screen.

A



B

```
example_cluster3.nii
example_cluster2.nii
example_cluster1.nii
example_cluster4.nii
```

The list above shows the nifti files you have uploaded in the "LOAD_NIFTI_FILE_HERE" folder. Copy name of nifti file you want to map (including the ".nii" termination) and paste it in the box below:

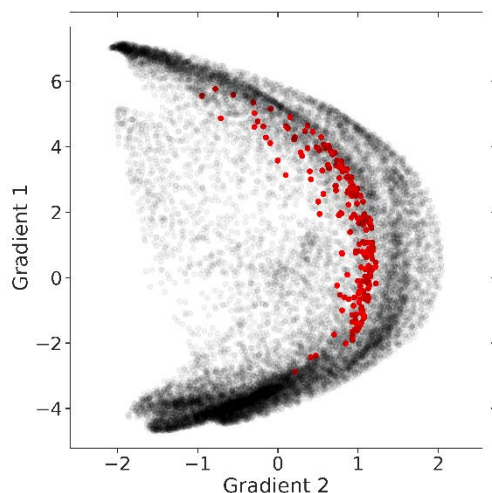
1st copy name of cluster you want to map here

Write a threshold number in the box below if you want to threshold your file; if no threshold leave blank:

2nd write threshold you want to use, or leave blank

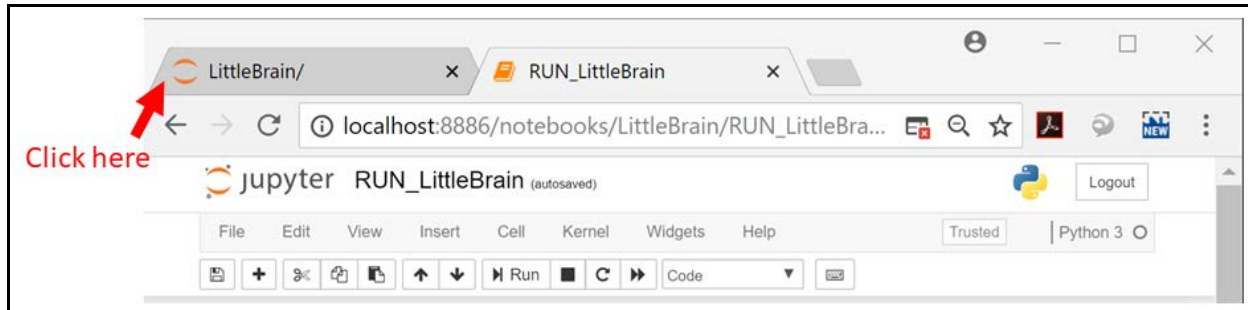
3rd click here

Output example:

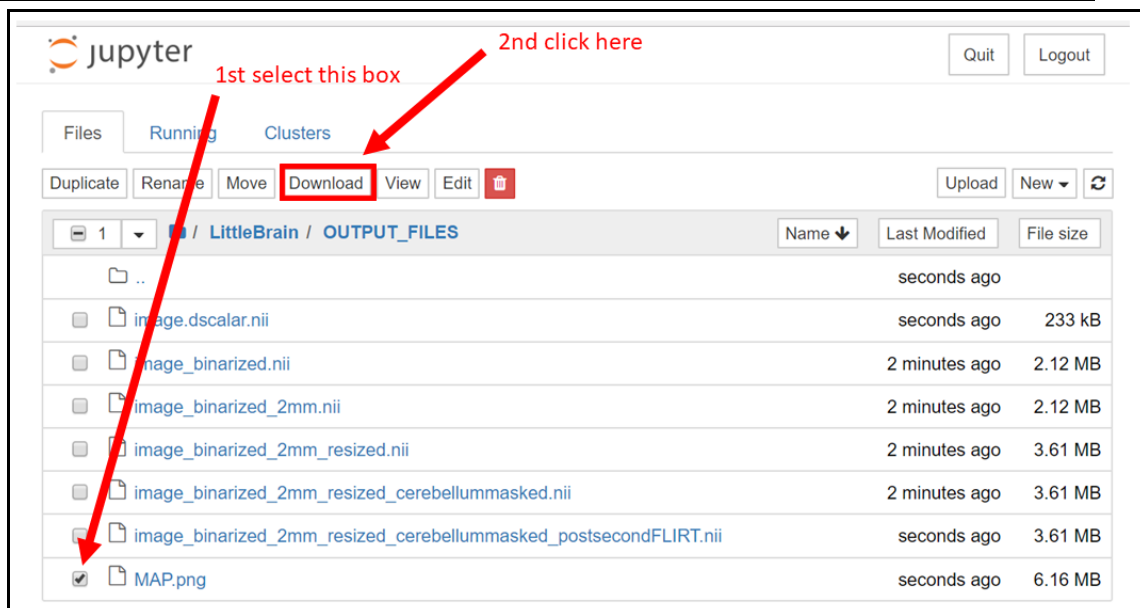
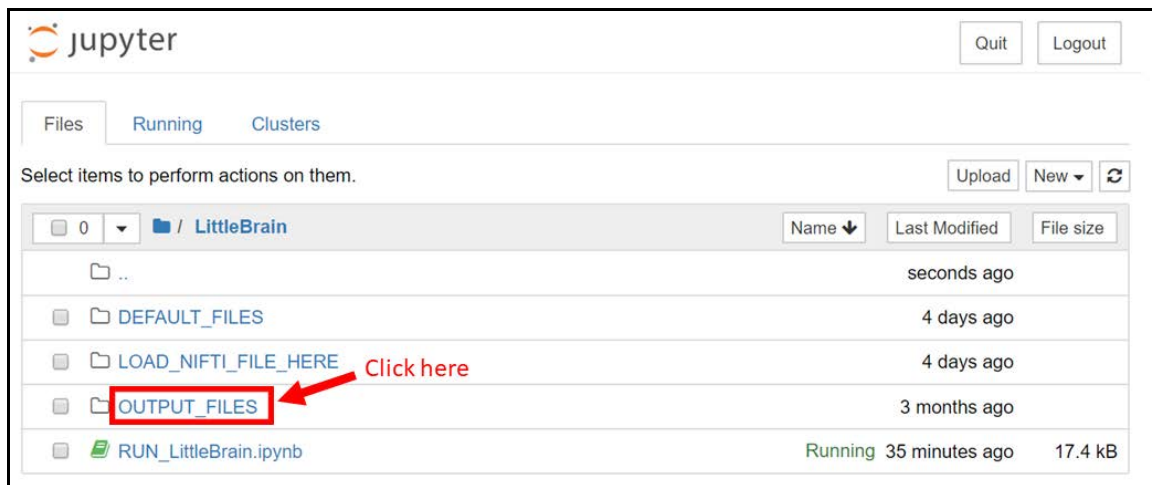




7. To download the scatterplot, go to the other tab that was opened in your browser:



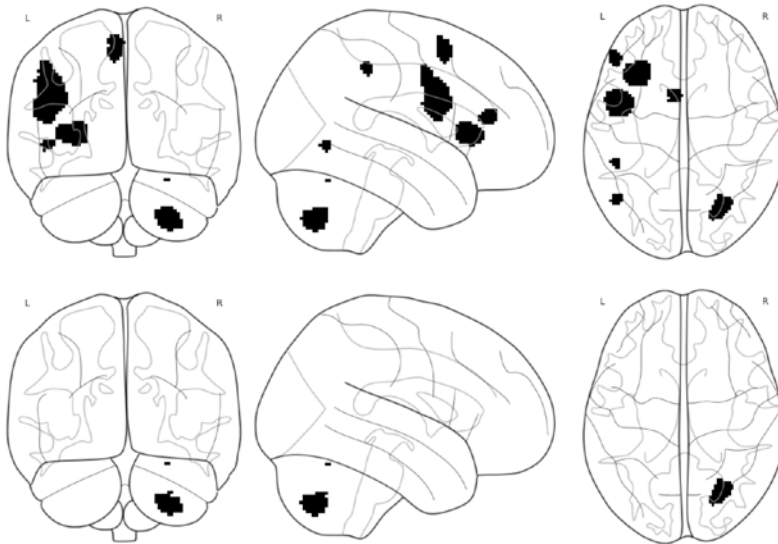
8. Go to the “OUTPUT_FILES” folder, and download “MAP.png”:



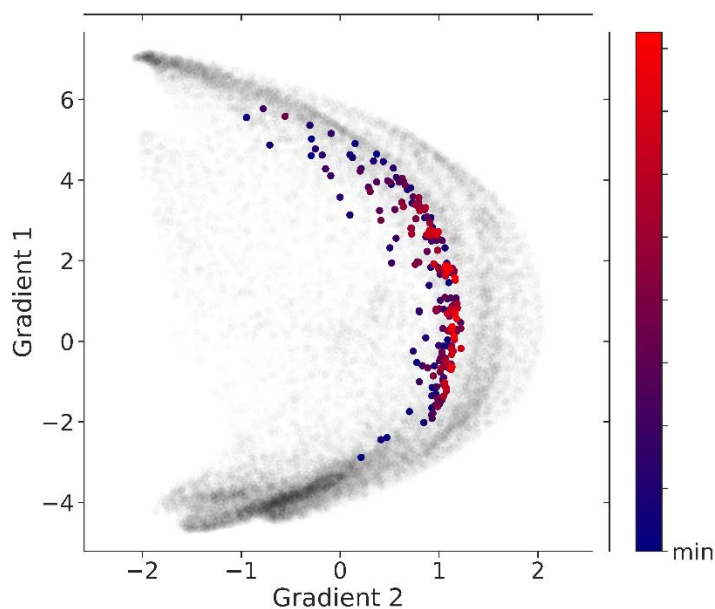


9. Additional options include:

- Check transformations (cerebellar masking, 2mm resolution change) that have occurred to the uploaded cerebellar data. This can be visualized by clicking the button “check transformations” after a map is generated. Output example:

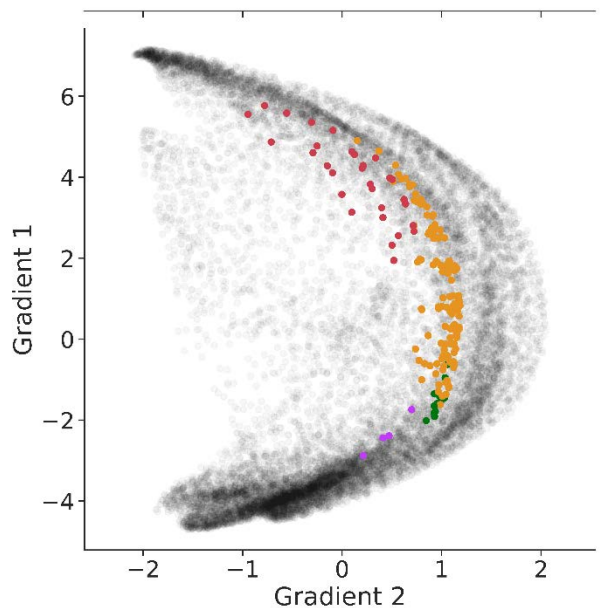


- Generate non-binary maps. These can be generated by clicking the button “see non-binary map” after a map is generated. Output example:





- Generate network-colored maps (where color of each dot corresponds to resting-state networks as defined in Buckner 2011 JNeurophysiology; blue=somatomotor network, purple = ventral attention network, green = dorsal attention network, cream = limbic network, orange = frontoparietal network, red = default-mode network, dark purple = visual network). These can be generated by clicking the button “see network-colored map” after a map is generated. Output example:



- Download Gradient 1 and Gradient 2 values for the clusters that have been mapped. These values can be downloaded from the “OUTPUT_FILES” folder as “GRADIENT1_values.csv” and “GRADIENT2_values.csv”.

If you are considering commercial use of FSL, please consult their license at <https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Licence>