

**allen and allen download**

---

Aran Thys Bauke van Hout

16-5-2022

## OVERVIEW

This document provides an overview of both the *allen* and the *allen\_download* function. In short, the *allen\_download* function allows you to easily download the tracer-data from the Allen Mouse Brain Connectivity Atlas (<https://mouse.brain-map.org/>), whereas the *allen* function enables you to interact with this data. Specially, *allen* provides an easy way to see which regions project to each other in the mouse brain.

In order to use *allen*, firstly the tracer-data has to be downloaded with *allen\_download*. Moreover, it is advised to run *allen* in the same directory as the one in which the downloaded tracer-data is located (this is not strictly necessary since you also could specify a path the tracer data, however if you are unfamiliar with python this might be complicated to do). Thus, I would advise to simply create a folder and run both *allen* and *allen\_download* in this folder.

## ALLEN DOWNLOAD

*allen\_download* downloads the tracer-data for the WT-experiments and exports the data as a single nifti file. In addition it also creates a cvs file with the experiment ids (WT\_experiment\_id\_list.csv), which is needed for the *allen* function. It accepts the following arguments:

- resolution: the resolution of the tracer-data that is to be downloaded (default=100).
- directory: the directory in which the data and the experiment\_id list is exported (default = current directory)
- brain\_mask: specifies whether the data outside of the brain should be converted to zero's. (default=true)

If you just run the function without specifying anything (*allen\_download()*). The function will simply proceed to download the tracer data at 100  $\mu$ M resolution with a brain mask, and export the data, together with the experiment\_id list to the current\_working directory.

## ALLEN

*allen* allows you to see which regions in the mouse brain project to each other. In addition it allows for the creation for masks and can provide an overview of all the relevant tracer experiments.

### ARGUMENTS;

*allen* accepts the following arguments;

- ROI\_anterograde
- ROI\_retrograde
- data
- activity\_treshold
- area\_treshold
- mask
- is\_injection
- overview
- resolution
- directory

### **ROI\_anterograde**

#### *General*

When a certain ROI\_anterograde is selected (eg: 'Secondary motor area'), the function will export all the tracer-experiments in which there was a primary injection in the ROI as nifti file. This will allow you to see to which other regions a certain ROI projects. The function does this by relying on the MouseConnectivityCache (MCC) provided by the Allen database itself. In essence, the function does nothing fancy here. It just provides a more user-friendly experience of interacting with the Allen brain database via the MCC. The list of the potential ROI's can be found here [http://api.brain-map.org/api/v2/structure\\_graph\\_download/1.json](http://api.brain-map.org/api/v2/structure_graph_download/1.json). By default the ROI\_anterograde=None

#### *In depth*

When a ROI is selected, the function will consult the Allen database via the MCC and retrieve all the experiment ID's with primary injection sites in the selected ROI. Subsequently, these ID's are converted to indices corresponding to downloaded tracer data. Through the use of these indices, a new nifti file with the relevant tracer-experiments is created and exported (see figure 1).

### Secondary motor area

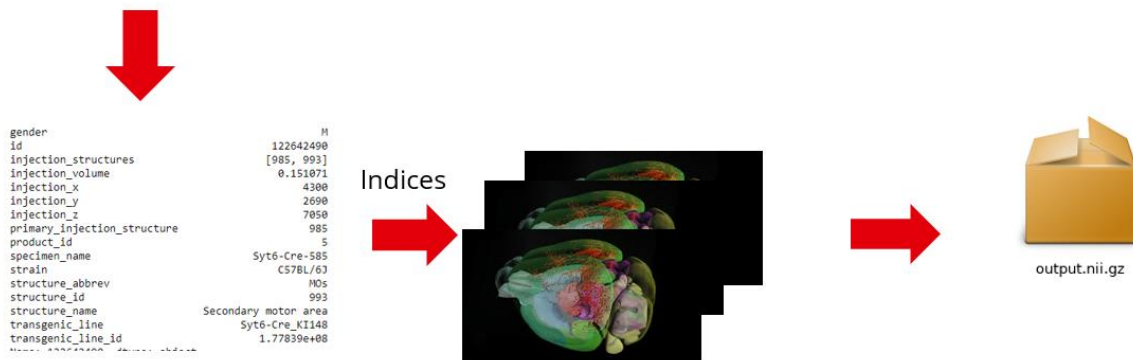


Figure 1

In order to convert the experiment ID's to indices, the function uses the `WT_experiment_list` file, which contains the ID's of the downloaded WT tracer data. It is important that the `WT_experiment_list` file is downloaded at the same time as the tracer-data itself (via `allen_download`). If this is not the case, the experiment ID's might no longer line up with the correct indices.

If the experiment ID file of the allen brain database is updated (eg; new experiments are added), the function will recognise this. It does this by comparing the downloaded experiment ID list with the up-to-date ID list from the allen database. If these lists don't match an error message will be displayed and the function will be terminated. However, this check will not be meaningful if the downloaded experiment ID list is up-to-date but the downloaded tracer-data is not. Hence, it is important that the tracer-data is downloaded at the same time as the experiment id file (`allen_download` makes sure this is the case).

### **ROI\_retrograde**

#### *General*

When a certain `ROI_retrograde` is selected, the function will export all the tracer-experiments in which there was thresholded tracer activity in the ROI into a nifti file. The list of the potential ROI's can be found here [http://api.brain-map.org/api/v2/structure\\_graph\\_download/1.json](http://api.brain-map.org/api/v2/structure_graph_download/1.json) (however, see disclaimer). This ability will allow you see which regions project to a certain ROI. By default the `ROI_retrograde=None`.

The function does this in 4 steps (see figure 2). First of all a mask is created for the ROI (eg infralimbic area) which is applied to downloaded tracer-data. Subsequently the tracer-activity values of the ROI are extracted and compared to a threshold for each experiment. If

they surpass the threshold the tracer-experiment is exported in the nifti file.

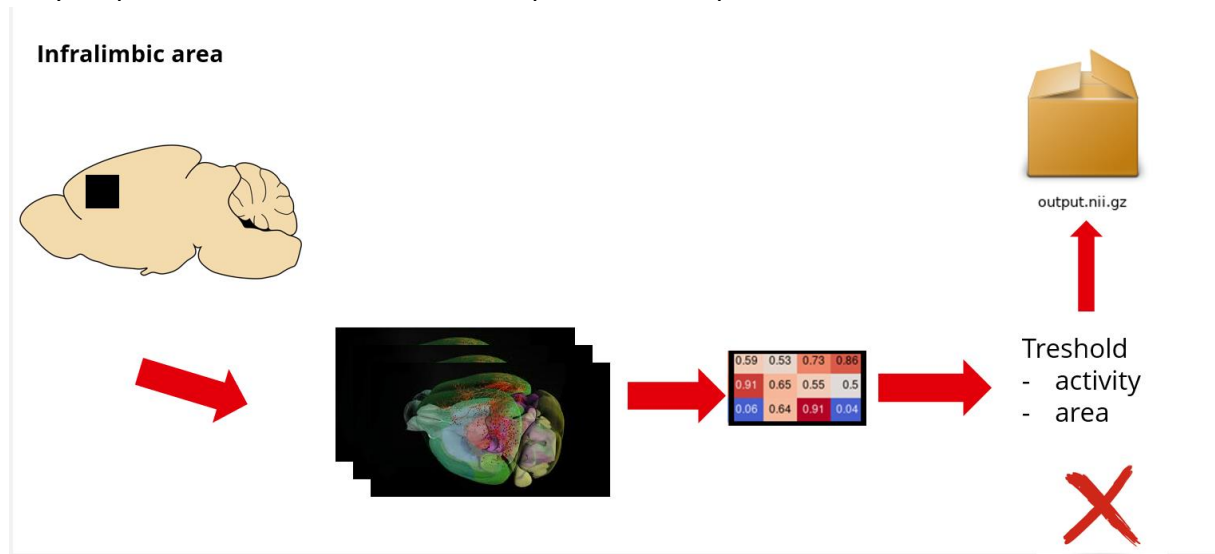


Figure 2

The threshold is determined by two factors, namely the `area_threshold` and the `activity_threshold`;

### **area\_threshold**

This threshold specifies the fraction of voxels in the ROI (eg 0.05) that have to match the `activity_threshold` in order for an experiment to be considered to have significant tracer activity. This threshold can vary anywhere between 0 and 1. By default it is set at 0.05.

### **activity\_threshold:**

This is the minimum tracer-activity that a voxel has to have in order to contribute to the `area_threshold`. By default it is set at 0.2.

### *In depth*

These two thresholds work together. For example, let's say the ROI consists of 1000 voxels. The `activity_threshold` determines the amount of tracer-activity that a voxel has to have in order to be considered to be significant (for example 0.1). The `area_threshold` subsequently determines how many voxels would have to meet this `activity_threshold` in order for the experiment to be considered significant. For example, If the `area_threshold` is set to 0.5, it would mean that 500 voxels in the ROI would have to meet the `activity_threshold` of 0.1 in order for the experiment to be considered significant and exported into the nifti file.

Putting everything together, the threshold works as follows:

- based upon the amount of voxels in the ROI\_retrograde, the `area_threshold` is quantified (eg 500 voxels)
- per experiment:
  - o the amount of voxels that match the activity threshold in the ROI is determined
  - o this amount is compared the quantified `area_threshold`

- when the amount of significantly activated voxels surpasses the `area_threshhold`, the `experiment_index` is appended to an experiment list
- the indices are used to get the relevant tracer-experiments from the tracer-data and export them into a single nifti file

### *Disclaimer*

In order to create a mask, the function interacts with the allen brain database via the MCC and the ReferenceSpacheCache. At first glance the allen brain database seems to allow masks to be created for all the regions on [http://api.brain-map.org/api/v2/structure\\_graph\\_download/1.json](http://api.brain-map.org/api/v2/structure_graph_download/1.json). However, for some small ROI's the ReferenceSpacheCache will export a faulty mask consisting of only zero's. *allen* checks this and prevents the export of these faulty masks and will instead produce a relevant error message.

### **data**

The data should take the form of numpy array loaded into python. It should consist of the WT tracer experiments of the allen brain database. This data can be downloaded and exported as a nifti file via *allen\_download*. In order to load the data into python, the following code can be run:

```
import numpy as np

import nibabel as nib

data= nib.load('tracer_data_masked.nii.gz')

data = np.array(data.dataobj)
```

### **mask**

#### *General*

The allen function is able to generate masks of the selected ROI's (ie, the ROI\_antergrade or ROI\_retrograde. These masks are then exported as nifti files. The mask can take two values (true/false). By default it is set at false.

#### *In depth*

It does this by interacting with the allen database itself, via the MCC and the ReferenceSpacheCache. However, see the *disclaimer* above.

### **resolution**

The resolution is set to 100  $\mu\text{m}$  by default. The allen function only works when the resolution is identical to the resolution of the downloaded tracer data. I would recommend to keep the resolution at 100  $\mu\text{m}$  since the function will take exponentially more time to run when the resolution is increased. Moreover, the memory load will also drastically increase. That being said, the resolution can be set to 100, 50, 25 or 10  $\mu\text{m}$ .

**directory**

The directory to which the nifti files are exported. This is set the current working directory by default.

**is\_injection**

This argument determines whether experiments for the retrograde tracer ability should include experiments in which the tracer was injected in the ROI itself. By default this is set to False.

**overview**

If this argument is set to True, an overview of the experiments that have been exported (for the ROI anterograde and the ROI retrograded) will be shown. This overview cannot be interacted with. By default the overview is set to False

**directory**

Here you can simply set the directory to which the nifti files should be exported.

## EXAMPLES

One robust way to use the allen function is by specifying all its arguments individually. You could just copy the code below and change the input of the arguments as you wish. In order to select a ROI, you would simply have to change None in a ROI (eg: 'infralimbic area').

```
allen(ROI_anterograde= None, ROI_retrograde=None, data=data, activity_threshold=0.2, area_threshold=0.05, mask=False, is_injection=False, overview=False, directory=None, resolution=100)
```

\*the directory=None simply means that the output-directory is set to the current directory.

However, it is often more convenient to specify certain arguments and leave others open. The arguments that are not specified will be defaulted to their default values. Following below are some examples. In general you will need at least 2 input arguments to make the function work, namely; Allen ('ROI\_anterograde/ROI\_retrograde', data=data)

```
allen()
```

- a help message with information is displayed

```
allen ('Infralimbic area', None, data):
```

- a nifti file for all the experiments with a primary injection site in the infralimbic area is created

```
allen(None, 'Prelimbic area', data, mask=True)
```

- a nifti file is created for all the experiments with thresholded (set at default) tracer activity in the prelimbic area (not including experiments in which there was a primary injection site in the prelimbic area). In addition a mask for the infralimbic area is created

```
allen (None, 'Prelimbic area', data, mask=True, is_injection=True)
```

- the same as before, but now also experiments are included in which there was a primary injection site in the prelimbic area

```
allen (None, 'Secondary motor area', data, activity_threshold=0.3)
```

- creates a nifti file for all the experiments with thresholded tracer activity in the secondary motor area at an activity\_threshold of 0.3

```
allen (None, 'Secondary motor area', data, area_threshold=0.1)
```

- creates a nifti file for all the experiments with thresholded tracer activity in the secondary motor area at an area threshold of 0.1

```
allen ('Anterior cingulate area', 'Anterior cingulate area, dorsal part', data, mask=True, overview=True, directory= 'home/example'):
```

- two nifti files are created, one with all the tracer data with a primary injection site in the anterior cingulate area, and one with all experiments with thresholded tracer



activity in the anterior cingulate area, dorsal part (not including experiments in which there was a primary injection site in the anterior cingulate area). Moreover, a mask is created for the anterior cingulate area and the anterior cingulate area, dorsal parts. Finally the nifti files are exported to a custom directory and an overview of all the relevant experiments is shown.