
Neuroscout-CLI

Neuroscout Team

Aug 02, 2022

CONTENTS

1	Contents	3
1.1	Installation	3
1.2	Usage	4
1.3	Command-Line Arguments	4
1.4	Outputs	7
	Index	9

The Neuroscout Command Line Interface (Neuroscout-CLI) allows you to easily execute analyses created on neuroscout.org. Neuroscout-CLI automatically fetches analysis dependencies (including data, and analysis specifications), fits a GLM model to the BIDS dataset, and produces shareable reports of the results.

Neuroscout-CLI uses [FitLins](#) to estimate linear models using the BIDS model specification.

Check out the [Usage](#) section for further information, including how to [Install](#) the project.

Note: If you are new to the Neuroscout project, visit the [Neuroscout](#) website and the official [Neuroscout Docs](#) for a general introduction.

CONTENTS

1.1 Installation

The recommended way to install `Neuroscout-CLI` is using containers (i.e. Docker or Singularity) to facilitate dependency management.

For demonstration, it is possible to run `Neuroscout-CLI` in the cloud (for free!) using Google Colab.

1.1.1 Containerized Execution

Docker

For most systems, we recommend using [Docker](#). First, follow the instructions for installing [Docker](#) on your system.

Next, follow the [Portable Docker Execution](#) guide in the Neuroscout Docs.

Singularity

[Singularity](#) containers are a great solution for High Performance Computing (HPC) environments, where *Docker* cannot typically be used due to more tightly controlled [user privileges](#).

First, check with your HPC administrator that *Singularity* is available for use. If so, follow our guide on [Singularity for HPCs](#) in the official [Neuroscout Docs](#).

1.1.2 Google Colab

To try Neuroscout without using any local resources, follow our interactive guide to [Cloud execution on Google Colab](#).

Google Colab allows you to execute Jupyter Notebooks for free, using two CPUs for several hours. This should be sufficient for individual Neuroscout analyses. A small demonstration can be run live in ~15 mins.

1.1.3 Manually prepared environment using pip

Danger

Manually installing *neuroscout-cli* can be difficult due to complex dependencies in the SciPy stack, or fMRI-specific tooling. Proceed only if you know what you're doing.

Use pip to install *neuroscout-cli* from PyPI:

```
pip install neuroscout-cli
```

1.2 Usage

Neuroscout-CLI executes models built using Neuroscout by acting as a lightway layer that fetches the data required for the model, and runs it using [FitLins](#). After the model is run, the results are outputs as BIDS Derivatives, and uploaded to NeuroVault (by default).

1.2.1 Containerized execution

Note that depending on your [Installation](#) method, the exact command will differ.

For **Docker**, you must prepend the command with `docker run -it` and map relevant local directories from the host to the container using `-v`. Instead of ``euroscout`, the command will be `neuroscout/neuroscout-cli` to reference a specific image. For example:

```
docker run -it -v LOCAL_DIR:OUT_DIR neuroscout/neuroscout-cli run ANALYSIS_ID OUT_DIR
```

For **Singularity**, you must prepend the command with `singularity run --cleanenv` and refer to a specific pre-downloaded image:

```
singularity run --cleanenv neuroscout-cli-<version>.simg ANALYSIS_ID OUT_DIR
```

For a complete guide, see [Portable Docker Execution](#) and [Singularity for HPCs](#) in the official [Neuroscout Docs](#).

1.3 Command-Line Arguments

1.3.1 neuroscout

Runs analyses created on neuroscout.org.

Neuroscout-CLI downloads the required data, configures outputs, and uses FitLins to execute analyses. Results are automatically uploaded to NeuroVault, facilitating data sharing.

In most use cases, the “run” command will handle all of the above, although the “get” and “upload” command are available for piecemeal execution.

Note: If using Docker, remember to map local volumes to the container using “-v” (such as OUT_DIR).


```
neuroscout [OPTIONS] COMMAND [ARGS] ...
```

get

Fetch analysis inputs.

Downloads the analysis bundle, preprocessed fMRI inputs, and configures output directory.

Inputs are downloaded to the output directory under *sourcedata*. If you run many analyses, you may wish to provide an *--download-dir* where datasets can be cached across analyses.

Note: *run* automatically calls *get* prior to execution, by default.

```
neuroscout get [OPTIONS] ANALYSIS_ID OUT_DIR
```

Options

--datalad-jobs <datalad_jobs>

Number of parallel jobs for DataLad when fetching files

--download-dir <download_dir>

Directory to cache input datasets, instead of OUT_DIR

--bundle-only

Only fetch analysis bundle, not imaging data

Arguments

ANALYSIS_ID

Required argument

OUT_DIR

Required argument

run

Run an analysis.

Automatically gets inputs and uploads results to NeuroVault by default.

This command uses FitLins for execution. Thus, any valid options can be passed through in [FITLINS_OPTIONS].

Note: *--model*, *--derivatives* and *--ignore* and positional arguments are automatically configured.

Example:

```
neuroscout run --force-upload --n-cpus=3 a54oo /out
```

If using Docker, remember to map local volumes to save outputs:

```
docker run --rm -it -v /local/dir:/out neuroscout/neuroscout-cli run a54oo /out
```

```
neuroscout run [OPTIONS] [FITLINS_OPTIONS] ... ANALYSIS_ID OUT_DIR
```

Options

- download-dir** <download_dir>
Directory to cache input datasets, instead of OUT_DIR
- datalad-jobs** <datalad_jobs>
Number of parallel jobs for DataLad when fetching files
- no-get**
Don't automatically fetch bundle & dataset
- upload-first-level**
Upload first-level results, in addition to group
- no-upload**
Don't upload results to NeuroVault
- fitlins-help**
Display FitLins help and options

Arguments

- FITLINS_OPTIONS**
Optional argument(s)
- ANALYSIS_ID**
Required argument
- OUT_DIR**
Required argument

upload

Upload results.

This command can be used to upload existing results to NeuroVault.

Note: *run* automatically calls *upload* after execution, by default.

```
neuroscout upload [OPTIONS] ANALYSIS_ID OUT_DIR
```

Options

- force-upload**
Force upload even if a NV collection already exists
- upload-first-level**
Upload first-level results, in addition to group

Arguments

ANALYSIS_ID

Required argument

OUT_DIR

Required argument

1.3.2 Optional FitLins arguments

Under the hood Neuroscout-CLI uses [FitLins](#) to execute the model. As such, Neuroscout-CLI will forward any arguments passed as `[FITLINS_OPTIONS]` to `__FitLins__`.

For details on valid FitLins arguments, please see [Usage](#).

1.4 Outputs

Neuroscout-CLI creates an output directory, with the name `neuroscout-ANALYSIS_ID` which contains both the inputs to the analysis (`sourcedata`), as well as the outputs of execution (`fitlins`).

Below is an example output directory.

```

/home/user/out/neuroscout-ANALYSIS_ID
├── sourcedata
│   ├── DATASET
│   │   └── fmriprep
│   ├── bundle
│   │   ├── events
│   │   ├── model.json
│   │   └── ...
├── fitlins
│   ├── sub-01
│   ├── reports
│   │   ├── dataset_description.json
│   │   └── ...
└── options.json

```

1.4.1 sourcedata directory

In the `sourcedata` folder, there are two folders: one containing the preprocessed fMRI inputs (the name of the folder is the name of the Dataset), and `bundle` which contains the contents of the analysis bundle for your `ANALYSIS_ID`.

.. note::

If specified `--download-dir` at run time (recomended; to cache the input directory in a common directory), you will not find the input data directory here.

Within the `bundle` directory you will find the event files and BIDS Stats Model (`model.json`) that are used to generate the design matrix for your analysis.

.. note::

For more information about **BIDS Stats Models**, take a look at the [official documentation](#).

1.4.2 fitlins directory

Within the `fitlins` directory, you will find the BIDS Derivatives compliant outputs from [FitLins](#) execution.

Within the `reports` folder, you can view interactive HTML reports, including a summary of your model, design matrices, and quality control visualizations.

1.4.3 Uploading to NeuroVault

By default, NeuroScout will upload all group and subject level results to NeuroVault, and update the NeuroScout API with the corresponding meta-data. You are free to opt out by specifying `--no-upload` at runtime.

Symbols

- bundle-only
 - neuroscout-get command line option, 5
- datalad-jobs
 - neuroscout-get command line option, 5
 - neuroscout-run command line option, 6
- download-dir
 - neuroscout-get command line option, 5
 - neuroscout-run command line option, 6
- fitlins-help
 - neuroscout-run command line option, 6
- force-upload
 - neuroscout-upload command line option, 6
- no-get
 - neuroscout-run command line option, 6
- no-upload
 - neuroscout-run command line option, 6
- upload-first-level
 - neuroscout-run command line option, 6
 - neuroscout-upload command line option, 6

A

ANALYSIS_ID

- neuroscout-get command line option, 5
- neuroscout-run command line option, 6
- neuroscout-upload command line option, 7

F

FITLINS_OPTIONS

- neuroscout-run command line option, 6

N

neuroscout-get command line option

- bundle-only, 5
- datalad-jobs, 5
- download-dir, 5
- ANALYSIS_ID, 5
- OUT_DIR, 5

neuroscout-run command line option

- datalad-jobs, 6
- download-dir, 6
- fitlins-help, 6

- no-get, 6
- no-upload, 6
- upload-first-level, 6
- ANALYSIS_ID, 6
- FITLINS_OPTIONS, 6
- OUT_DIR, 6

neuroscout-upload command line option

- force-upload, 6
- upload-first-level, 6
- ANALYSIS_ID, 7
- OUT_DIR, 7

O

OUT_DIR

- neuroscout-get command line option, 5
- neuroscout-run command line option, 6
- neuroscout-upload command line option, 7