

BIDS and its tools, MRIQC and fMRIPrep

Yann Leprince

NeuroSpin / UNIACT

14 February 2023



What is BIDS?



- ▶ A standard for organizing neuroimaging and behavioral data
- ▶ Facilitate data **reuse** and **sharing**
- ▶ Compatible with many **analysis tools**

Seminal paper

Gorgolewski, K. J. et al. The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. Scientific Data 3, 160044 (2016).

Heavily inspired by the format used by the OpenNeuro repository (formerly known as OpenfMRI).

Raw data, derived data, source data

Raw data

BIDS is primarily for **raw data** (*minimally* pre-processed data: file format conversion, anonymization)

Derived data

Derivatives of the raw data must be kept separate from the raw data.

Source data

Examples: DICOM files, E-Prime files...

Recommended file structure:

```
my_dataset/  
  sourcedata/  
  rawdata/  
    dataset_description.json  
    participants.tsv  
    sub-01/  
    sub-02/  
  derivatives/  
    pipeline_1/  
    pipeline_2/
```

Raw data **must** adhere to the spec

Each derivative may be BIDS-compliant dataset

Overview of the directory structure

`sub-control01/` ← One directory for each subject

`anat/` ← One directory for each data type

`sub-control01_T1w.nii.gz`

`sub-control01_T1w.json`

`sub-control01_T2w.nii.gz`

`sub-control01_T2w.json`

`dwi/`

`sub-control01_dwi.nii.gz`

`sub-control01_dwi.bval`

`sub-control01_dwi.bvec`

`fmap/`

`sub-control01_phasediff.nii.gz`

`sub-control01_phasediff.json`

`sub-control01_magnitude1.nii.gz`

`code/`

`deface.py`

`participants.tsv` ← Describes properties of participants (age, sex, handedness...)

`dataset_description.json` ← Dataset-level metadata (name, licence, authors...)

README

CHANGES

Principles of file naming

```
sub-ed110159_task-localizer_run-02_bold.nii.gz
```

The name of a data file is a concatenation of:

1. a chain of `<key>-<value>` *entities* separated by underscores (_).
2. a *suffix* that usually designates the modality
Other examples: `_T1w`, `_T2w`, `_bold`, `_dwi`, `_defacemask`
3. an *extension* that identifies the file format

General rule

Every file in a BIDS dataset must be identified uniquely by its name alone, i.e. even when taken out of its containing directory.

- ▶ The ordering of entities is fixed by the standard
- ▶ For each file type, entities can be required, optional, or disallowed

Multi-session data

In the case of multi-session (longitudinal) data, a *session* directory level is added, and the `ses-<label>` entity becomes required:

```
sub-control01/  
  ses-predrug/  
    func/  
      sub-control01_ses-predrug_task-nback_bold.nii.gz  
      sub-control01_ses-predrug_task-nback_bold.json  
      sub-control01_ses-predrug_task-nback_events.tsv  
      sub-control01_ses-predrug_task-nback_sbref.nii.gz  
    sub-control01_ses-predrug_scans.tsv  
  ses-postdrug/  
    func/  
      sub-control01_ses-postdrug_task-nback_bold.nii.gz  
      sub-control01_ses-postdrug_task-nback_bold.json  
      sub-control01_ses-postdrug_task-nback_events.tsv  
      sub-control01_ses-postdrug_task-nback_sbref.nii.gz  
    sub-control01_ses-postdrug_scans.tsv  
  sub-control01_sessions.tsv ← Describe key variables that change between sessions (age...)
```

File formats

BIDS mandates the use of a few file formats:

- ▶ NIfTI for images (.nii.gz is recommended)
- ▶ TSV (tab-separated values) for tabular data
 - ▶ `participants.tsv` contains the list of subjects and their characteristics (age, sex, etc.)
 - ▶ `_events.tsv` for functional stimuli, responses, etc.
 - ▶ `_physio.tsv.gz` for cardiac, respiratory recordings, etc.
- ▶ JSON (JavaScript Object Notation) for key-value metadata

Metadata is stored in
sidecar JSON files,
for example:

```
sub-control01_T1w.nii.gz  
sub-control01_T1w.json
```

```
{  
  "Modality": "MR",  
  "MagneticFieldStrength": 3,  
  "ManufacturersModelName": "Prisma_fit",  
  "InstitutionName": "NeuroSpin",  
  "EchoTime": 0.00305,  
  "RepetitionTime": 2.3,  
  "InversionTime": 0.9,  
  "FlipAngle": 9,  
  "PartialFourier": 0.875,  
  ...  
}
```

Data types

These *data types* are currently specified by BIDS:

- ▶ anat
- ▶ dwi
- ▶ fmap
- ▶ func
- ▶ perf
- ▶ eeg
- ▶ ieeg
- ▶ meg
- ▶ pet
- ▶ beh
- ▶ micr

For each of these *data types*, required and optional files are specified in the standard.

Evolution of the BIDS standard

BIDS is an evolving standard, developed in the spirit of an open-source project.

BEPs: BIDS Enhancement Proposals

`bids.neuroimaging.io/get_involved`

Examples:

- ▶ BEP001: Quantitative MRI (integrated in v1.5.0)
- ▶ BEP012: Functional preprocessing derivatives
- ▶ BEP022: Magnetic resonance spectroscopy
- ▶ ...

The current version of the standard is v1.8.0, released 29 October 2022.

- ▶ Minor release will contain backwards-compatible changes;
- ▶ Potentially disruptive changes will be implemented in a future 2.0 version of BIDS.

bids-validator

BIDS Validator v1.5.6

bids_dataset

Summary	Available Tasks	Available Modalities
<ul style="list-style-type: none">31 Files, 815.44MB2 - Subjects1 - Session	<ul style="list-style-type: none">localizer	<ul style="list-style-type: none">T1wsbrf

Your dataset is not a valid BIDS dataset.

[view 1 error in 16 files](#)

Error 1: [Code 1] NOT_INCLUDED 16 files

[Click here for more information about this issue](#)

Files with such naming scheme are not part of BIDS specification. This error is most commonly caused by typos in file names that make them not BIDS compatible. Please consult the specification and make sure your files are named correctly. If this is not a file naming issue (for example when including files not yet covered by the BIDS specification) you should include a ".bidsignore" file in your dataset (see <https://github.com/bids-standard/bids-validator#bidsignore> for details). Please note that derived (processed) data should be placed in derivatives folder and source data (such as DICOMS or behavioural logs in proprietary formats) should be placed in the /sourcedata folder.

[view 1 warning in 1 file](#)

[Download error log for bids_dataset](#)

[Click to view details on BIDS specification](#)

If you have any questions please post on [Neurostars](#)

The source code for the validator can be found [here](#)

`bids-standard.github.io/bids-validator/`

```
yl243478@is234203:~/bids_dataset
yl243478@is234203:~/bids_dataset $ bids-validator .
bids-validator@1.5.3

This dataset appears to be BIDS compatible.

Summary:          Available Tasks:    Available Modalities:
35 Files, 217.99MB
4 - Subjects
2 - Sessions
FLAIR
T1w

If you have any questions, please post on https://neurostars.org/tags/bids.

yl243478@is234203:~/bids_dataset $
```

Installed on NeuroSpin workstations

Creating a BIDS dataset

Data are transmitted by the scanners in DICOM format, and need to be converted to BIDS.

- ▶ **neurospin_to_bids** is specifically adapted for importing data acquired on the NeuroSpin platform (3 T, 7 T, soon 11,7 T, also MEG)
 - ▶ Accessible on NeuroSpin workstations as `neurospin_to_bids`
 - ▶ `github.com/neurospin/neurospin_to_bids`
 - ▶ Define the list of subjects and sequences to import in a `participants_to_import.tsv` file
- ▶ External tools are also available e.g. `dcm2bids`
 - ▶ `unfmontreal.github.io/Dcm2Bids/`
- ▶ Both tools use `dcm2nii` for the DICOM-to-Nifti conversion and the extraction of JSON metadata

BIDS tools for data analysis

Some tools were designed from the start to work with BIDS-formatted data:

- ▶ MRIQC: quality control of MRI acquisitions (structural and BOLD)
- ▶ fMRIPrep: preprocessing pipeline for fMRI

BIDS-Apps are containerized versions of other processing pipelines: bids-apps.neuroimaging.io

BIDS-Apps/freesurfer version v6.0.1-6.1

BIDS-Apps/ndmg version v0.1.0

BIDS-Apps/BROCCOLI version v1.0.1

BIDS-Apps/FibreDensityAndCrosssection version v0.0.1

BIDS-Apps/SPM version v0.0.20

poldracklab/mriqc version v0.15.3

BIDS-Apps/QAP version v0.0.1

BIDS-Apps/CPAC version v1.0.1a_22

BIDS-Apps/hyperalignment version v0.0.5

BIDS-Apps/mindboggle version v0.0.4-1

BIDS-Apps/MRtrix3_connectome version v0.4.2

BIDS-Apps/rs_signal_extract version v0.1

BIDS-Apps/aa version v0.2.0

General principles of Quality Control (QC)

Evaluation of the quality of raw images

- ▶ Visual inspection
 - ▶ fundamentally important for knowing one's data
 - ▶ the eye is sensitive to certain types of defects, not others
 - ▶ exhaustive inspection is too time-consuming
 - ⇒ sampling (random or guided)
- ▶ Visualization tools
 - ▶ Speed up visual inspection
 - ▶ Highlight defects to make them visible to the eye
- ▶ Detection tools
 - ▶ Detection of “suspicious” data to be visually inspected

Evaluation of the quality of data processing

- ▶ Very dependent on the kind of processing
- ▶ General idea: inspect intermediate results in order

Storage of quality control results

Quality control table:

- ▶ One line per subject / session / image
- ▶ Result column: e.g. **0** for “pass”, **1** for “exclude”
- ▶ Comments column for keeping notes

Efforts have been made to standardize the storage of QC results at the level of NeuroSpin (Édouard Duchesnay, Antoine Grigis, Josselin Houenou).

If you do a QC, you need to save it in the /derivatives folder with a filename format as follows :

modality_date_qc.tsv

modality : cat12-version_vbm, rawdata, morphologist, quasi-raw....

The date field is optional, you need it if you have several versions of your QC.

MRIQC

MRIQC is a tool that helps with quality control of raw data:

- ▶ structural MRI (T1w, T2w)
- ▶ functional MRI (bold)

Original paper

Esteban, O. et al.

MRIQC: Advancing the automatic prediction of image quality in MRI from unseen sites.

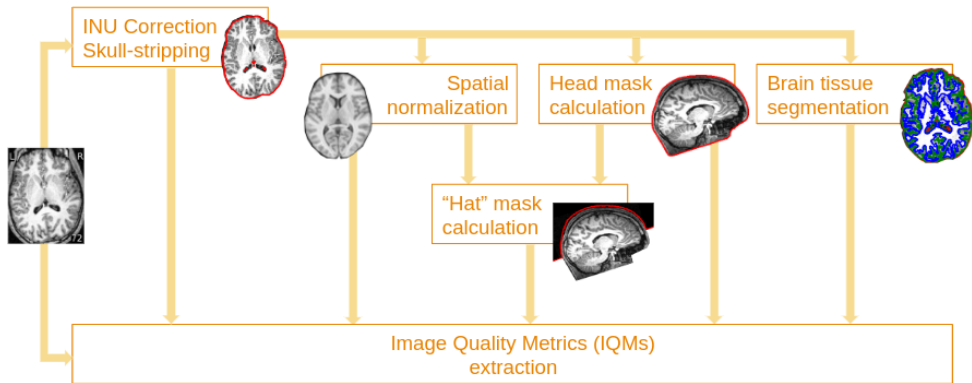
PLOS ONE 12, e0184661 (2017). DOI: [10.1371/journal.pone.0184661](https://doi.org/10.1371/journal.pone.0184661).

Derived from QAP Quality Assessment Protocol of the PCP project (Preprocessed Connectomes Project).

- ▶ Extraction of (Image Quality Metrics, IQMs)
- ▶ Individual visual reports and group reports

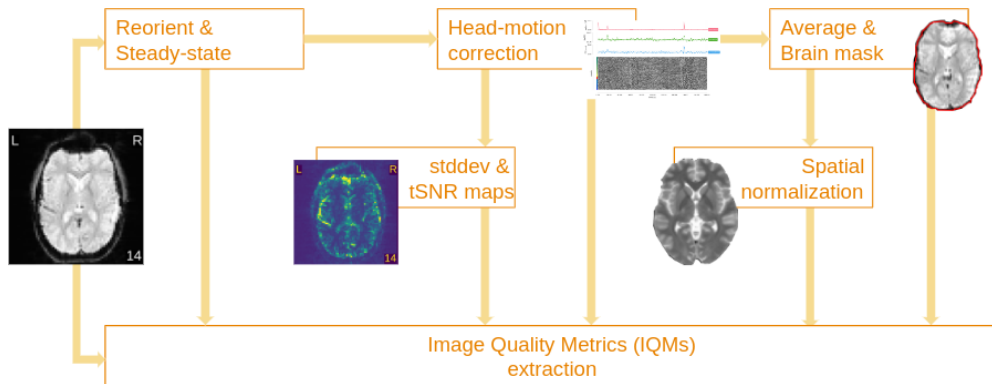
Quality metrics of anatomical images

- ▶ noise-based metrics (CJV, CNR, SNR, SNRd, Qi2)
- ▶ information-theoretic metrics (EFC, FBER)
- ▶ metrics targeting specific artefacts (INU, Qi1, wm2max)
- ▶ other metrics (FWHM, rpve, icvs, overlap)



Quality metrics of functional images

- ▶ metrics of spacial information (EFC, FBER, SNR)
- ▶ metrics of temporal information (DVARs, gcor, tSNR)
- ▶ metrics targeting specific artefacts (FD, gsr)
- ▶ other metrics (aor, aqi, dummy)

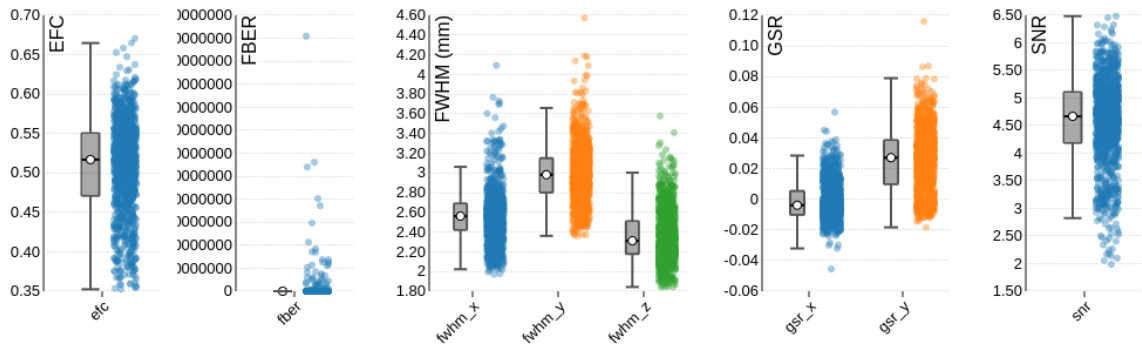


The MRIQC group report

MRIQC: group bold report

Summary

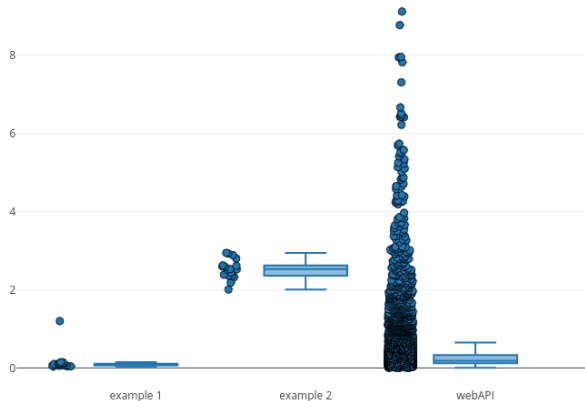
- Date and time: 2018-06-12, 16:18.
- MRIQC version: 0.11.0+3.g836bf07.dirty.



...

mriqcception

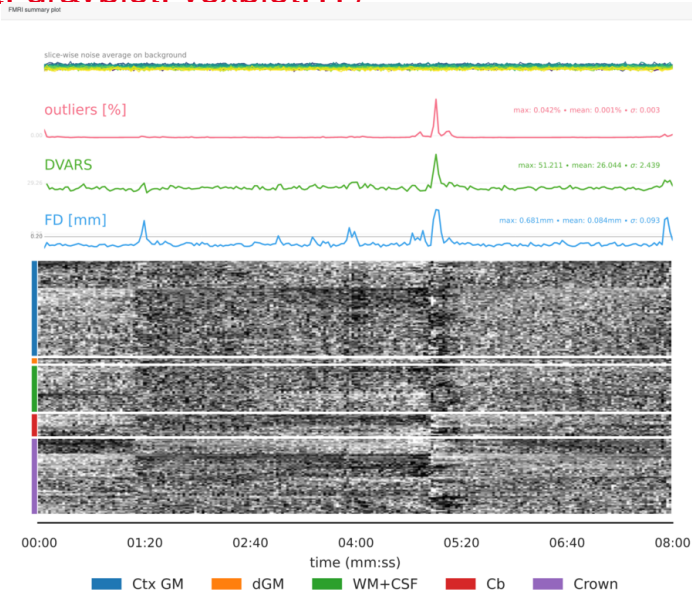
- ▶ A way of comparing your dataset against a crowdsourced collection of anonymized metrics submitted by MRIQC users
- ▶ <https://github.com/elizabethbeard/mriqcception>



Carpetplot (a.k.a. staticplot. aravplot. voxplot. . .)

Original paper

Power, J. D. A simple but useful way to assess fMRI scan qualities. NeuroImage 154, 150–158 (2017). DOI: 10.1016/j.neuroimage.2016.08.009



How to run MRIQC

- ▶ The recommended way of running MRIQC is to use the container image (a.k.a. Docker image) through Singularity
- ▶ The image is fetched with:

```
export SINGULARITY_TMPDIR=/volatile/tmp/...
export SINGULARITY_CACHEDIR=/volatile/tmp/...
singularity pull docker://nipreps/mriqc:22.0.6
```
- ▶ singularity pull needs a large amount of disk space (typically $\approx 10 - 30$ GB) and the default temporary directory /tmp is too small

```
tmpdir=$(mktemp -d --tmpdir=/volatile/tmp \
    mriqc.XXXXXXXXXX)
timeout --kill-after=1m 2h \
    singularity run \
        --cleanenv \
        --bind /path/to/rawdata:/rawdata:ro \
        --bind /path/to/derivatives/mriqc-22.0.6:/out
        --bind "$tmpdir":/tmpdir \
        /path/to/mriqc_22.0.6.sif \
        --work-dir /tmpdir \
        --verbose-reports \
        /rawdata /out participant
rm -rf "$tmpdir"
```

How to run MRIQC

- ▶ BIDS-Apps such as MRIQC run in a *container* that is isolated from your system
- ▶ Arguments before `mriqc_*.sif` are for Singularity
- ▶ Arguments that follow `mriqc_*.sif` are for MRIQC
- ▶ Directories must be *bound* into the container:

`--bind <host_path>:<container_path>:[ro|rw]`

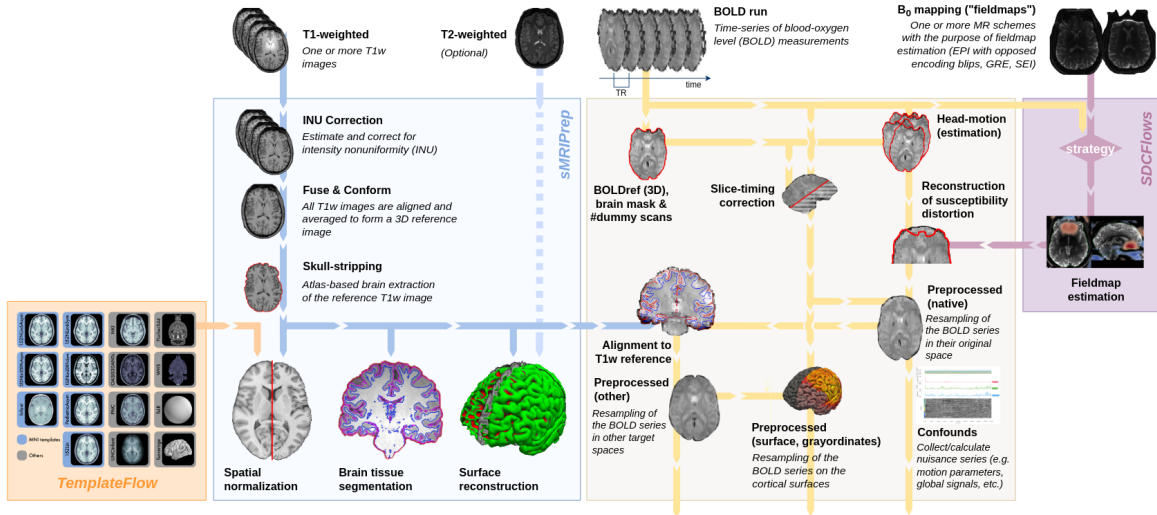
```
tmpdir=$(mktemp -d --tmpdir=/volatile/tmp \  
    mriqc.XXXXXXXXXX)  
timeout --kill-after=1m 2h \  
    singularity run \  
    --cleanenv \  
    --bind /path/to/rawdata:/rawdata:ro \  
    --bind /path/to/derivatives/mriqc-22.0.6:/out  
    --bind "$tmpdir":/tmpdir \  
    /path/to/mriqc_22.0.6.sif \  
    --work-dir /tmpdir \  
    --verbose-reports \  
    /rawdata /out participant  
rm -rf "$tmpdir"
```

How to run MRIQC

- ▶ mriqc does not clean up its temporary files: the disk can be filled quickly
- ▶ mriqc can sometimes crash, in that case the remaining images will not be processed
⇒ write a batch script to launch one instance on each subject, using
--participant-label, --session-id, and/or --modalities
- ▶ mriqc can sometimes become stuck in an infinite loop: use the timeout command (processing takes \approx 15min per image so the 2h timeout includes a safety margin)

```
tmpdir=$(mktemp -d --tmpdir=/volatile/tmp \  
    mriqc.XXXXXXXXXX)  
timeout --kill-after=1m 2h \  
    singularity run \  
    --cleanenv \  
    --bind /path/to/rawdata:/rawdata:ro \  
    --bind /path/to/derivatives/mriqc-22.0.6:/out  
    --bind "$tmpdir":/tmpdir \  
    /path/to/mriqc_22.0.6.sif \  
    --work-dir /tmpdir \  
    --verbose-reports \  
    /rawdata /out participant  
rm -rf "$tmpdir"
```

The fMRIPrep pipeline



Recommended way to run fMRIPrep

```
singularity run --cleanenv \  
  --bind /i2bm/local/freesurfer/license.txt:/freesurfer-license.txt:ro \  
  --bind /path/to/rawdata:/rawdata:ro \  
  --bind /path/to/derivatives/fmriprep-22.1.1:/out:rw \  
  --bind /volatile/tmp/fmriprep:/tmpdir:rw \  
  /volatile/opt/fmriprep_22.1.1.sif --skip_bids_validation \  
  --work-dir=/tmpdir --clean-workdir --fs-license-file=/freesurfer-license.txt \  
  /rawdata /out participant
```

Resources

- ▶ BIDS Specification: see `bids.neuroimaging.io`
- ▶ BIDS Validator: `bids-standard.github.io/bids-validator/`
- ▶ `github.com/neurospin/neurospin_to_bids`
- ▶ `unfmontreal.github.io/Dcm2Bids/`
- ▶ `mriqc.org`
- ▶ `https://github.com/elizabethbeard/mriqception`
- ▶ `fmriprep.org`

Hands-on session

Option 1: get MRIQC to run under Singularity

- ▶ Your own dataset in BIDS format
- ▶ We can provide a test dataset, an extract from the IBC dataset (Individual Brain Charting)
- ▶ Please do not saturate the Wi-Fi with `singularity pull`, we can distribute the containers on USB sticks

Option 2: exploring a real-world MRIQC Group report

- ▶ Group report of the ABIDE dataset:
`mriqc.s3.amazonaws.com/abide/bold_group.html`
- ▶ Look at the definition and values of the different Image Quality Metrics, notice which one are most sensitive to different artefacts
- ▶ Go to `mriqc.org` for the documentation of Image Quality Metrics (IQMs)