**Pybids:** Python tools for manipulation and analysis of BIDS datasets

**Tal Yarkoni**¹, Alejandro de la Vega², Elizabeth DuPre³, Oscar Esteban³, Yaroslav O. Halchenko⁴, Michael Hanké⁵, Valerie Hayot-Sasson⁶, Alexander Ivanov⁷, Gregory Kiar¹, Christopher Markiewicz³, Quinten McNamara³, Dmitry Petrov⁶, Taylor Salo, Dylan Nicholson⁷, Jean-Baptiste Poline⁴, Russell Poldrack⁶, Krzysztof Gorgolewskii⁸

¹University of Texas at Austin, ²McGill University, ³Stanford University, ⁴Dartmouth College, and several other fine institutions

---

**Introduction**

- Efforts to standardize the representation of neuroimaging datasets have recently converged on the Brain Imaging Data Structure (BIDS; Gorgolewski et al., 2016)—a relatively simple specification that has already been adopted by hundreds of researchers around the world
- To maximize the utility of this common standard, it is important to develop easy-to-use tools that facilitate programmatic interaction with, and manipulation of, BIDS-compliant datasets.
- Here we describe a new open-source Python package—"pybids"—that provides powerful tools for querying BIDS datasets and constructing complex statistical analysis pipelines.

---

**Methods**

We wrote a bunch of code

---

**Why should I use it?**

- Makes it much easier to work with BIDS-compliant datasets in Python
- Querying module ("grabbids") makes it easy to construct and execute complex queries that might otherwise require extensive scripting.
- Loading/extraction of all variables (task events, physiological recordings, behavioral measures, etc.) found in BIDS projects—optionally returned as pandas DataFrames
- A working implementation of the forthcoming BIDS-Model specification, which provides a simple, machine-readable way to represent complex statistical models that can potentially be fitted using a variety of fMRI analysis packages
- Partial auto-generation of methods sections
- Hundres dozens: three other features

---

**Overflow space for comment box below**

---

**Where can I get it?**

[https://github.com/incf/pybids](https://github.com/incf/pybids)

Hipsters please point your mobile devices at the magic square on the right

---

**What does it do?**

- Python package ("pybids") that provides powerful tools for querying BIDS datasets and constructing complex statistical analysis pipelines.

---

**About pybids**

**How do I use it?**

**Initialize a BIDS project**

```python
from bids import BIDSLayout
layout = BIDSLayout(bids_dir)
```

**Simple but flexible querying**

```python
# Get list of unique subjects
layout.get_subjects()
```

```python
# Get filenames of all BOLD timeseries for subject '04'
layout.get(type='bold', subject='04', return_type='file')
```

```python
# Get metadata for all images
layout.get_metadata(target='sub-04_task-rest_run-1_bold.nii.gz')
```

**Easy access to variables in BIDS projects**

```python
# Load all run-level variables (e.g., experimental events,ophysio recordings, etc.) in the BIDS project
events = layout.get_collections('level-run')
```

---

**Model specification and design matrix construction**

Pybids supports the BIDS-Model specification; here’s an example JSON file specifying a simple first-level model:

```json
{
  "name": "my_model",
  "blocks": {
    "name": "run",
    "level": "run",
    "transformations": {
      "name": "scale",
      "inputs": "level",
      "trial_type": [{
        "parametric gain": "parametric",
        "parametric loss": "B1",
        "auto contrasts": true
      }]
    }
  }
}
```

We can pass this specification directly to pybids, and have it easily construct design matrices, contrast specifications, etc.—potentially after applying complex transformations to the variables included in the design matrix.

```python
from bids import Analysis
analysis = Analysis(layout, model_json)
```

**And more...**

There are thousands of things you can do with pybids. More information at [https://github.com/incf/pybids](https://github.com/incf/pybids)

---

**Comments, suggestions, requests, sketches, haiku, and other expressions of [dis]satisfaction go in this box**