

Nipype: Neuroimaging analysis pipeline and interfaces in Python.

Problem

- Multiple analysis packages, each with their own pipelining functionality and language
- Each package has a steep learning curve

Ideally

- Mix-and-match components from different packages
- Compare different packages
- Iterate over a parameter set
- Provenance tracking

Solution

- Nipype: A common framework for interfacing multiple analysis packages, and for elegant pipelining.
- <http://nipy.sourceforge.net/nipype/>

```
from nipype.interfaces.spm import Realign, Coregister
from nipype.interfaces.fsl import Bet
...
rlgn = NodeWrapper(interface=Realign())
rlgn.inputs.register_to_mean = True

coreg = NodeWrapper(interface=Coregister())

skullstrip = NodeWrapper(interface=Bet())
```

```
ds = NodeWrapper(interface=DataSource())
ds.inputs.base_directory = '~/data/exp001'
...
pline = pe.Pipeline()
pline.connect([(ds, rlgcn, [(\`func\`, \`infile\`)]),
               (rlgcn, coreg, [(\`mean_image\`, \`source\`)]),
               ...
               (coreg, skullstrip, [(\`coreg_files\`...])

pline.run()
```