

HTC and Cognitive Neuroscience

OSG User School Spotlight

Christopher Cox

University of Wisconsin—Madison

July 31, 2015

My motivation

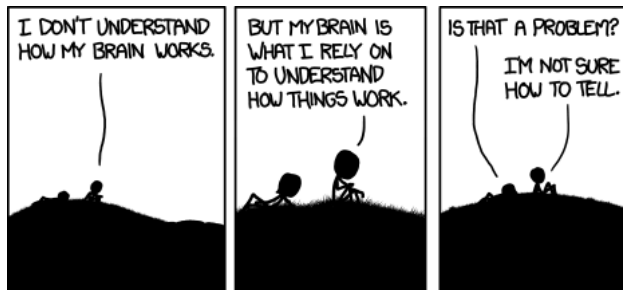
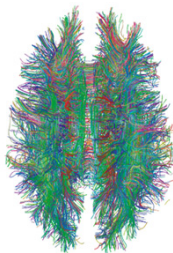


Figure: xkcd 1163

HTC and
Cognitive
Neuroscience

- ▶ I am an aspiring Cognitive Neuroscientist.



- ▶ I am an aspiring Cognitive Neuroscientist.
- ▶ I think of the brain as a vast, complicated network that utilizes **distributed representations**.





Why I use HTCondor

Some definitions

Mental representations Psychological elements like concepts, thoughts, and emotions.

Why I use HTCondor

Some definitions

Mental representations Psychological elements like concepts, thoughts, and emotions.

Neural Representations Physical states of the brain that implement mental representations.

Why I use HTCondor

Some definitions

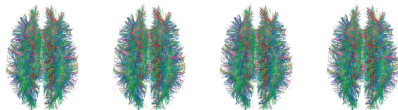
Mental representations Psychological elements like concepts, thoughts, and emotions.

Neural Representations Physical states of the brain that implement mental representations.

Distributed representation A pattern of neural activity composed of units with varying response profiles. The units may not be well localized in the brain. The same units may encode many representations by activating in different patterns.

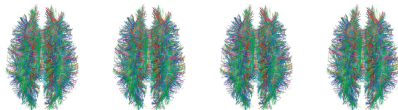
Why I use HTCondor

- ▶ fMRI data consists of 3D snapshots of the brain collected through time.



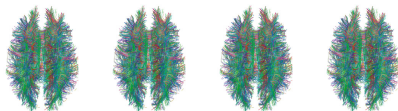
Why I use HTCondor

- ▶ fMRI data consists of 3D snapshots of the brain collected through time.
- ▶ A full snapshot is collected every two seconds.



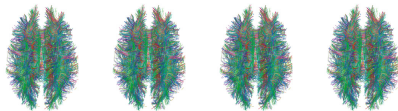
Why I use HTCondor

- ▶ fMRI data consists of 3D snapshots of the brain collected through time.
- ▶ A full snapshot is collected every two seconds.
- ▶ Each snapshot consists of tens of thousands of voxels that fall within the skull.



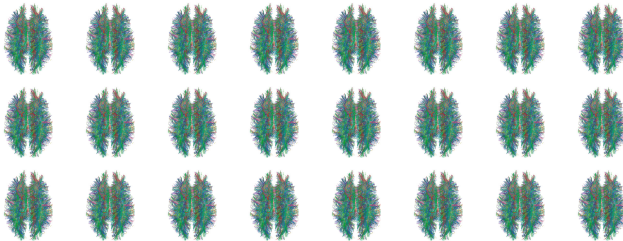
Why I use HTCondor

- ▶ fMRI data consists of 3D snapshots of the brain collected through time.
- ▶ A full snapshot is collected every two seconds.
- ▶ Each snapshot consists of tens of thousands of voxels that fall within the skull.
- ▶ The challenge is to identify a set of voxels that, when taken together, can be used to discriminate between different mental representations.



... so, *why* use HTC?

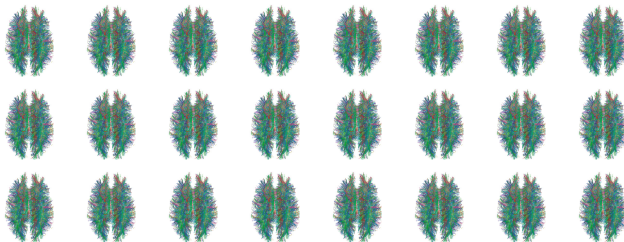
- We began with hypotheses about how the brain might actually work, and designed new algorithms that reflected those hypotheses.¹



¹SOS Lasso (Rao, Cox, Nowak & Rogers, NIPS 2013)

... so, *why* use HTC?

- ▶ We began with hypotheses about how the brain might actually work, and designed new algorithms that reflected those hypotheses.¹
- ▶ (But they are much more computationally intensive.)



¹SOS Lasso (Rao, Cox, Nowak & Rogers, NIPS 2013)

How intensive?

- ▶ A single job may take anywhere from 30 minutes to 3 hours to complete, depending on the hyper-parameter configuration.

How intensive?

- ▶ A single job may take anywhere from 30 minutes to 3 hours to complete, depending on the hyper-parameter configuration.
- ▶ Modest cross-validated parameter sweeps can have 10,000 jobs, amounting to roughly a year's worth of processing.

How intensive?

- ▶ A single job may take anywhere from 30 minutes to 3 hours to complete, depending on the hyper-parameter configuration.
- ▶ Modest cross-validated parameter sweeps can have 10,000 jobs, amounting to roughly a year's worth of processing.
- ▶ Experimenting with these algorithms would be impossible without HTC.

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).
- ▶ To main throughput, one must:

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).
- ▶ To main throughput, one must:
 1. Parse and summarize the results (efficiently and thoroughly).

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).
- ▶ To main throughput, one must:
 1. Parse and summarize the results (efficiently and thoroughly).
 2. Make decisions (may require collaboration).

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).
- ▶ To main throughput, one must:
 1. Parse and summarize the results (efficiently and thoroughly).
 2. Make decisions (may require collaboration).
 3. Launch a new batch of jobs.

Workflow: Maintaining throughput

- ▶ The challenge becomes keeping pace with the results coming back (now I am the bottleneck).
- ▶ To maintain throughput, one must:
 1. Parse and summarize the results (efficiently and thoroughly).
 2. Make decisions (may require collaboration).
 3. **Launch a new batch of jobs.**

A (partial) workflow solution

Job and DAG setup can be **automated** in a way that is **easy to maintain** and **intuitive to work with** using file templating.

Requirements

1. *Basic* working knowledge of Perl. (I'm proof that you don't need to know much.)
2. Getting comfortable with JSON and/or YAML.
3. The Text::Template module for Perl.

- ▶ Each file that you need to generate many times will need a template.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.
 3. Represent those parameters as a key-value based data structure (a Perl “hash”).

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.
 3. Represent those parameters as a key-value based data structure (a Perl “hash”).
 4. Pass that hash into the template to fill it in.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.
 3. Represent those parameters as a key-value based data structure (a Perl “hash”).
 4. Pass that hash into the template to fill it in.
 5. Write the filled template to a file.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.
 3. Represent those parameters as a key-value based data structure (a Perl “hash”).
 4. Pass that hash into the template to fill it in.
 5. Write the filled template to a file.

- ▶ Each file that you need to generate many times will need a template.
- ▶ Each template will be associated with a short Perl program that will (at minimum):
 1. Read the template, and prepare it for filling.
 2. Read a parameter file.
 3. Represent those parameters as a key-value based data structure (a Perl “hash”).
 4. Pass that hash into the template to fill it in.
 5. Write the filled template to a file.

(This is the same way some dynamic web pages are constructed.)

It takes all kinds

- ▶ I do not have a background in computer science or engineering (but I see the value and have committed to learning).

It takes all kinds

- ▶ I do not have a background in computer science or engineering (but I see the value and have committed to learning).
- ▶ I do not have petabytes of data to process and explore (but that does not mean HTCCondor is overkill—quite the contrary).

It takes all kinds

- ▶ I do not have a background in computer science or engineering (but I see the value and have committed to learning).
- ▶ I do not have petabytes of data to process and explore (but that does not mean HTCCondor is overkill—quite the contrary).
- ▶ HTC has rapidly become essential to my work.

It takes all kinds

- ▶ I do not have a background in computer science or engineering (but I see the value and have committed to learning).
- ▶ I do not have petabytes of data to process and explore (but that does not mean HTCCondor is overkill—quite the contrary).
- ▶ HTC has rapidly become essential to my work.
- ▶ It is possible to do a great deal without extensive prior knowledge.

Thank you

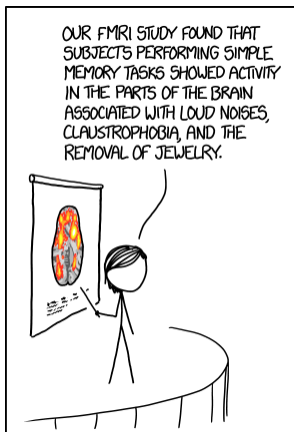


Figure: xkcd 1453

<https://github.com/crcox/condortools> (for the intrepid...)