

Characterizing and Optimizing Generalizability and Sensitivity in Neuroimaging Through Repeated Analysis

Greg Kiar

2018-2021 NSERC Alexander Graham Bell Fellow,
2017 McGill University Healthy Brains for Healthy Lives Fellow,
McGill Centre for Integrative Neuroscience, Montreal Neurological Institute,
Ph.D. student McGill University,
M.S.E. Johns Hopkins University,
B.Eng Carleton University

Outline

- Background & Overview
- Chapter 1: Accessible and Repeatable Scientific Computing
- Chapter 2: Characterization of pipeline generalizability and sensitivity
- Chapter 3: Pipeline optimization in varying inference contexts
- Conclusion (& demos, time permitting)

Background & Overview

Many publicly available MRI datasets

- ADNI
- ABCD
- ABIDE
- ADHD-200
- Age-ility
- AIBL
- BRAINS
- CamCAN
- CMI-HBN
- COBRE
- CoRR/FCP-INDI
- DLBS
- fBIRN
- GSP
- HCP
- IXI
- Kirby21
- MASSIVE
- MindBoggle-101
- MIRIAD
- MPI-LMBB
- MSC
- NACC
- NCANDA
- NKIRS
- OASIS-CS
- OASIS-Long
- OpenfMRI
- PING
- PNC
- PTBP
- SALD
- SchizConnect
- StudyForrest
- UK-Biobank

Source: <https://github.com/cMadan/openMorph>

And, many supported BIDS apps

- AFNI
- ANTS Cortical Thickness
- Baracus
- Brainiak-srm
- BROCCOLI
- CPAC
- DPARSF
- Fibre Density and
- Cross-section
- fMRIprep
- Freesurfer
- FSL Tools
- HCP Pipelines
- Hyper Alignment
- MAGeTbrain
- MindBoggle
- MRIQC
- MRtrix3 Connectome
- ndmg
- NIAK
- OPPNI
- SRM
- SPM
- Tracula
- QAP

Source: <http://bids-apps.neuroimaging.io/apps/>

Reproducibility is a measurable problem

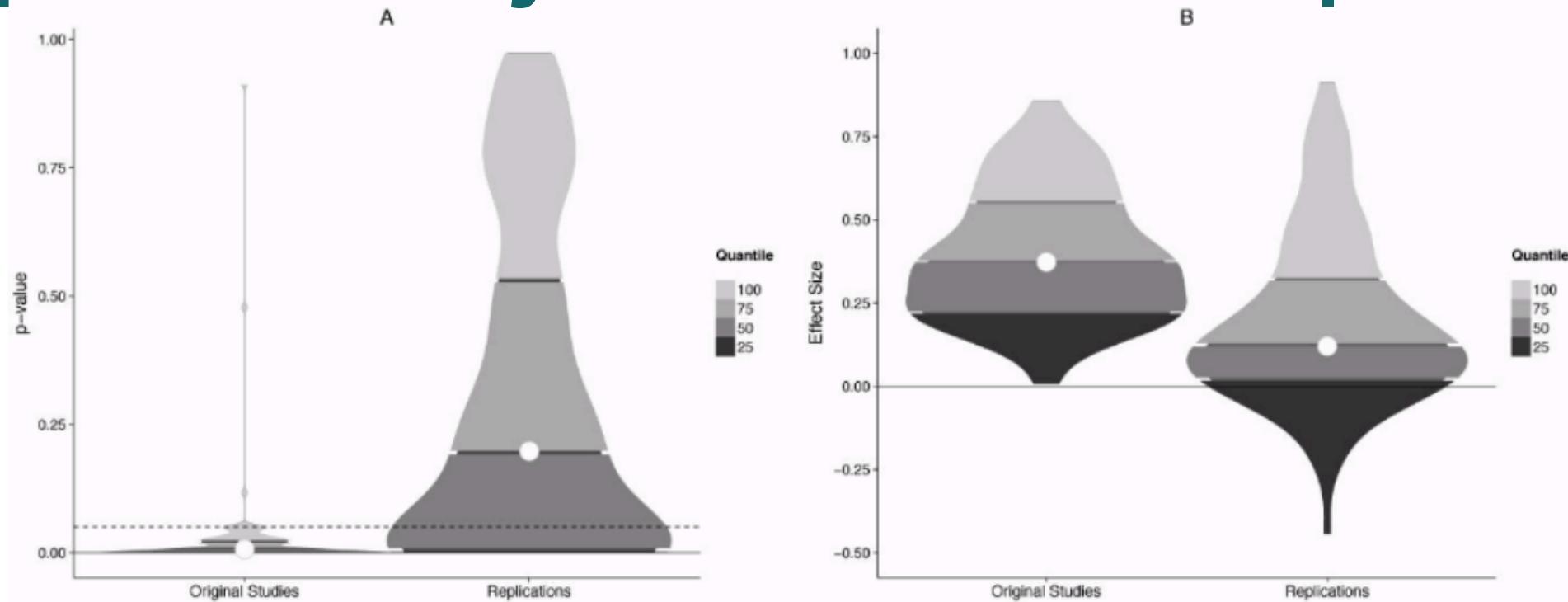


Fig. 1. Density plots of original and replication *P* values and effect sizes. (A) *P* values. (B) Effect sizes (correlation coefficients). Lowest quantiles for *P* values are not visible because they are clustered near zero.

(Open Science Collaboration, 2015)

Neuroscience is no exception

- Software bugs have led to inflated false-positive rates (Eklund et al., 2016)
- Operating system differences have led to different results (Glatard et al., 2015)
- Similar tools performing similar operations give different results (Bowring, 2018)
- Ignoring variability in data quality leads to different results (Khundrakpam et al., 2017)
- Neuroscience studies tend to be under-sampled and have low power (Button, 2013)
- Noisy data and incomplete statistics can lead to spurious results (Bennett et al., 2011)

A common approach to neuroscience

1. Pose a hypothesis
2. Collect + curate dataset
3. Manually perform QC on dataset
4. Pick processing pipeline and parameters
5. Process random subset with pipeline in 4.
6. Manually perform QC on derivatives
7. Redo from 4. if not happy with 6.
8. Process all data with pipeline in 4.
9. Answer statistical question
10. Publish claim

A common worst-case approach to neuroscience

1. Pose a hypothesis
2. Collect + curate dataset
3. Manually perform QC on dataset
4. Pick processing pipeline and parameters
5. Process random subset with pipeline in 4.
6. Manually perform QC on derivatives
7. Redo from 4. if not happy with 6.
8. Process all data with pipeline in 4.
9. Answer statistical question
10. Publish claim

Skipped or expedited
← (Khundrakpam et al. , 2017) vs. (Haar et al., 2016)

A common worst-case approach to neuroscience

1. Pose a hypothesis
2. Collect + curate dataset
3. Manually perform QC on dataset
4. Pick processing pipeline and parameters
5. Process random subset with pipeline in 4.
6. Manually perform QC on derivatives
7. Redo from 4. if not happy with 6.
8. Process all data with pipeline in 4.
9. Answer statistical question
10. Publish claim

Skipped or expedited
← (Khundrakpam et al. , 2017) vs. (Haar et al., 2016)

← Arbitrarily chosen (i.e. defaults)

A common worst-case approach to neuroscience

1. Pose a hypothesis
2. Collect + curate dataset
3. Manually perform QC on dataset
4. Pick processing pipeline and parameters
5. Process random subset with pipeline in 4. ← Skipped or expedited
(Khundrakpam et al. , 2017) vs. (Haar et al., 2016)
6. Manually perform QC on derivatives
7. Redo from 4. if not happy with 6.
8. Process all data with pipeline in 4. ← Arbitrarily chosen (i.e. defaults)
9. Answer statistical question
10. Publish claim ← Homogeneous sub-sample

A common worst-case approach to neuroscience

1. Pose a hypothesis
 2. Collect + curate dataset
 3. Manually perform QC on dataset
 4. Pick processing pipeline and parameters
 5. Process random subset with pipeline in 4.
 6. Manually perform QC on derivatives
 7. Redo from 4. if not happy with 6.
 8. Process all data with pipeline in 4.
 9. Answer statistical question
 10. Publish claim
- Skipped or expedited
← (Khundrakpam et al., 2017) vs. (Haar et al., 2016)
- Arbitrarily chosen (i.e. defaults)
← Homogeneous sub-sample
- ← Prone to overfitting, undocumented

A common worst-case approach to neuroscience

1. Pose a hypothesis
 2. Collect + curate dataset
 3. Manually perform QC on dataset
 4. Pick processing pipeline and parameters
 5. Process random subset with pipeline in 4.
 6. Manually perform QC on derivatives
 7. Redo from 4. if not happy with 6.
 8. Process all data with pipeline in 4.
 9. Answer statistical question
 10. Publish claim
- ← Skipped or expedited
← (Khundrakpam et al. , 2017) vs. (Haar et al., 2016)
- ← Arbitrarily chosen (i.e. defaults)
- ← Homogeneous sub-sample
- ← Prone to overfitting, undocumented
- ← Heterogeneous collection, no provenance

A common worst-case approach to neuroscience

1. Pose a hypothesis
 2. Collect + curate dataset
 3. Manually perform QC on dataset
 4. Pick processing pipeline and parameters
 5. Process random subset with pipeline in 4.
 6. Manually perform QC on derivatives
 7. Redo from 4. if not happy with 6.
 8. Process all data with pipeline in 4.
 9. Answer statistical question
 10. Publish claim
- Skipped or expedited
← (Khundrakpam et al., 2017) vs. (Haar et al., 2016)
- Arbitrarily chosen (i.e. defaults)
← Homogeneous sub-sample
- Prone to overfitting, undocumented
← Heterogeneous collection, no provenance
- Of questionable reproducibility
(Open Science Collaboration, 2015)

I plan to address this, with:

1. Creation of infrastructures for accessible and reproducible neuroimaging analysis.
2. The development of methods and models to characterize pipeline generalizability.
3. The a priori optimization of context-specific pipeline selection in neuroimaging.

Chapter 1: Accessible and Repeatable Scientific Computing

1 year

The issue is...

Existing platforms in neuroscience such as CBRAIN¹, OpenNeuro², and others, enable running standard tools on configured environments with ingested datasets from the web.

No known platform in this space enables all of

- development of tools,
- on arbitrary data,
- without a persistent service running,
- on a variety of computational infrastructures,
- programmatically.

Source 1: <https://portal.cbrain.mcgill.ca/login>

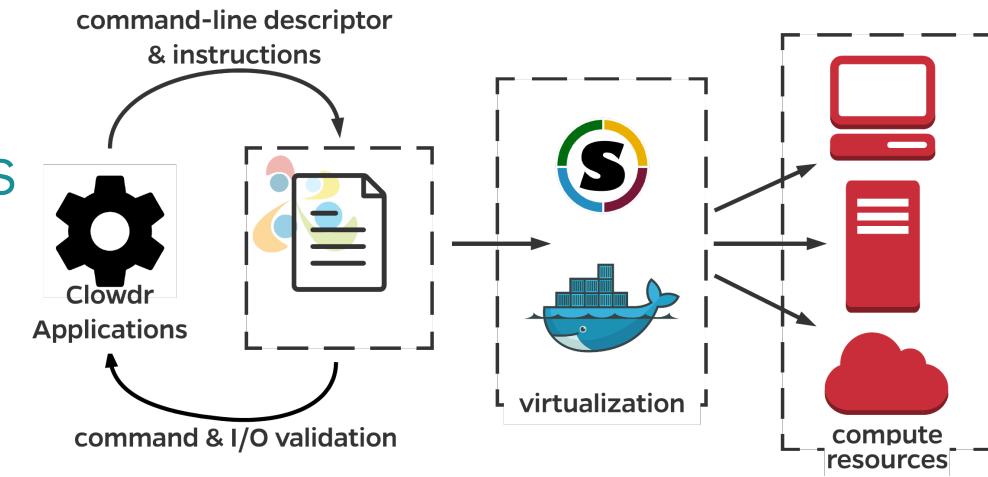
Source 2: <https://openneuro.org/>

Two key challenges

- Server-less deployment and re-execution of neuroimaging analyses
- Enabling the discovery and construction of heterogeneous datasets

Clowdr

- Rapidly iterate on developing pipelines
- Easily sweep hyperparameter settings
- Accessibly share task progress, status, and outputs



Clowdr Share

Session Info

Tool	BIDS-example
Session Location	examples/task/bids-example/clowdr/

Executions

#	Invocation	Date Launched	Status																																								
0	invocation_sub-01.json	Mar 28, 2018 (11:00:45)	Complete																																								
<table><tr><td>Property</td><td>Value(s)</td><td></td><td></td></tr><tr><td>Task Files</td><td>Task Definition, Tool Descriptor, Invocation</td><td></td><td></td></tr><tr><td>Input Data Location</td><td>/data/ds114/sub-01</td><td></td><td></td></tr><tr><td>Output Data Location</td><td>/Users/greg/code/gkiar/clowdr/examples/task/2018-03-22-CA56LMEI/BIDS-example</td><td></td><td></td></tr><tr><td>Tool Inputs</td><td></td><td>Logs</td><td></td></tr><tr><td>BIDS directory</td><td>/data/ds114</td><td>Output Log, Error Log</td><td></td></tr><tr><td>Output directory name</td><td>/data/ds114/derivatives/bids-example/</td><td></td><td></td></tr><tr><td>Analysis level</td><td>participant</td><td>Task Duration</td><td></td></tr><tr><td>Participant label</td><td>['01']</td><td>0:00:16.738347</td><td></td></tr><tr><td>Skip validator</td><td>True</td><td></td><td></td></tr></table>				Property	Value(s)			Task Files	Task Definition, Tool Descriptor, Invocation			Input Data Location	/data/ds114/sub-01			Output Data Location	/Users/greg/code/gkiar/clowdr/examples/task/2018-03-22-CA56LMEI/BIDS-example			Tool Inputs		Logs		BIDS directory	/data/ds114	Output Log, Error Log		Output directory name	/data/ds114/derivatives/bids-example/			Analysis level	participant	Task Duration		Participant label	['01']	0:00:16.738347		Skip validator	True		
Property	Value(s)																																										
Task Files	Task Definition, Tool Descriptor, Invocation																																										
Input Data Location	/data/ds114/sub-01																																										
Output Data Location	/Users/greg/code/gkiar/clowdr/examples/task/2018-03-22-CA56LMEI/BIDS-example																																										
Tool Inputs		Logs																																									
BIDS directory	/data/ds114	Output Log, Error Log																																									
Output directory name	/data/ds114/derivatives/bids-example/																																										
Analysis level	participant	Task Duration																																									
Participant label	['01']	0:00:16.738347																																									
Skip validator	True																																										
1	invocation_sub-02.json	Mar 28, 2018 (11:00:45)	Complete																																								
2	invocation_sub-03.json	Mar 28, 2018 (11:00:45)	Complete																																								

Analysis with Clowndr

```
$ # Installable on Python3...
$ pip install clowndr
$ # Run locally, on clusters, the cloud, and share results
$ clowndr local {tool} {invocation} {dataset} {output loc}
$ clowndr cluster {tool} {invocation} {dataset} {output loc} {scheduler}
$ clowndr cloud {tool} {invocation} {dataset} {output loc} {cloud} {keys}
$ clowndr share {task loc} # {task loc} returned by any of the above
$
```

Apine

- Web interface for exploring and discovering remote datasets
- Query datasets by feature, such as:
 - modality,
 - number of participants,
 - scanner type,
 - task type, etc.

JSON Schema

```
{  
  "$schema": "http://json-schema.org/draft-04/schema#",  
  "$id": "https://github.com/INCF/apine/blob/master/apine.schema.json",  
  "type": "object",  
  "additionalProperties": false,  
  "properties": {  
    "dataset": {  
      "$id": "https://github.com/INCF/apine/blob/master/apine.schema.json/dataset",  
      "minLength": 1,  
      "format": "uri-reference",  
      "type": "string",  
      "title": "dataset",  
      "description": "The parent"  
    },  
    "participant": {  
      "$id": "https://github.com/INCF/apine/blob/master/apine.schema.json/participant",  
      "minLength": 1,  
      "type": "string",  
      "title": "participant",  
      "description": "The corresponding participant"  
    },  
    "session": {  
      "$id": "https://github.com/INCF/apine/blob/master/apine.schema.json/session",  
      "minLength": 1,  
      "type": "string",  
      "title": "session",  
      "description": "The corresponding session"  
    },  
    "modality": {  
      "$id": "https://github.com/INCF/apine/blob/master/apine.schema.json/modality",  
      ...  
    }  
  }  
}
```

JSON Representation

```
{  
  "dataset": "ds001",  
  "participant": "01",  
  "modality": "anat",  
  "filename": "sub-01_inplaneT2.nii.gz",  
  "filename_keys": [  
    "inplaneT2"  
  ]  
}
```

REST API

GET	/datasets	Returns list of dataset IDs
GET	/dataset	Returns list of dataset contents
GET	/modalities	Returns list of modalities
GET	/modality	Returns list of images matching the modality
GET	/participants	Returns list of participants
GET	/participant	Returns list of images matching the participant
GET	/sessions	Returns list of sessions
GET	/session	Returns list of images matching the session
GET	/filename_keys	Returns list of filename keys

Query collections with Apine

```
$ # Available on Github  
$ git clone git@github.com:INCF/apine.git ./apine  
$ cd ./apine/swagger/python-flask-server-generated/  
$  
$ # Launch server to enable querying  
$ python -m swagger_server  
$  
$ # Query datasets in the web
```

ⓘ localhost:8080/gkiar/apine-dev/0.0.1/dataset?datasetID&modality=func,anat&filename_key=inplaneT2,T1w,run-02

Chapter 2: Characterization of Pipeline Generalizability and Sensitivity

1.5 years (total: 2.5 years)

The issues are...

- Tool selection is often more arbitrary than principled
- There is limited understanding as to what tools perform well, when
- There is no general method for characterizing and comparing tools

Previous reliable optimization strategies

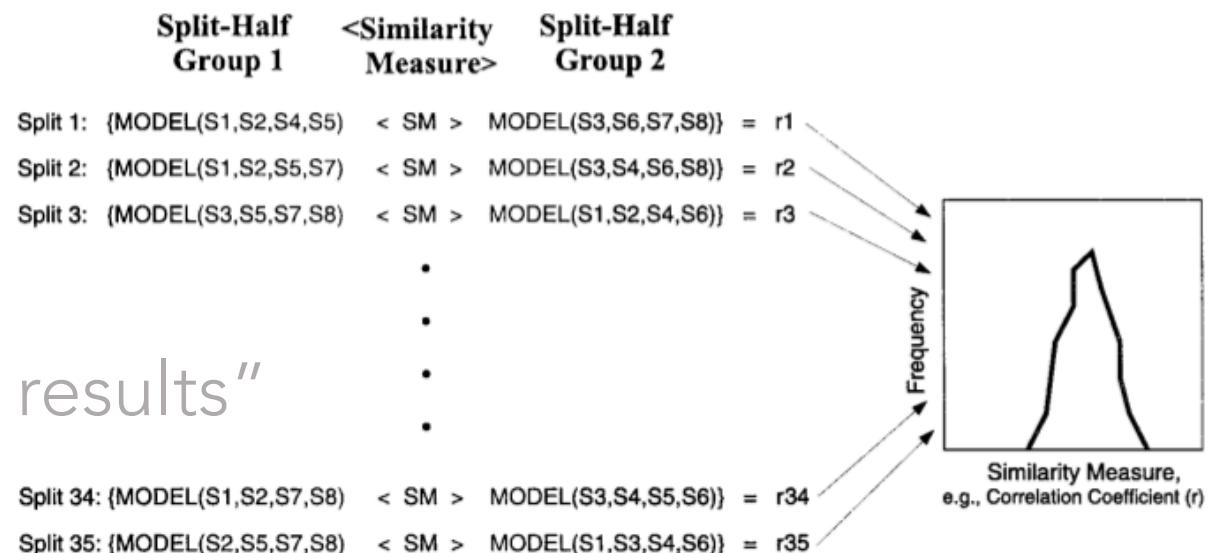
- Discriminability (Kiar et al., 2018)

$$D = p(|g_{ij} - g_{ij'}| \leq |g_{ij} - g_{i'j''}|)$$

“the probability that two observation of the same things are more similar than two observations of two different things”

- NPAIRS (Strother et al., 2002)

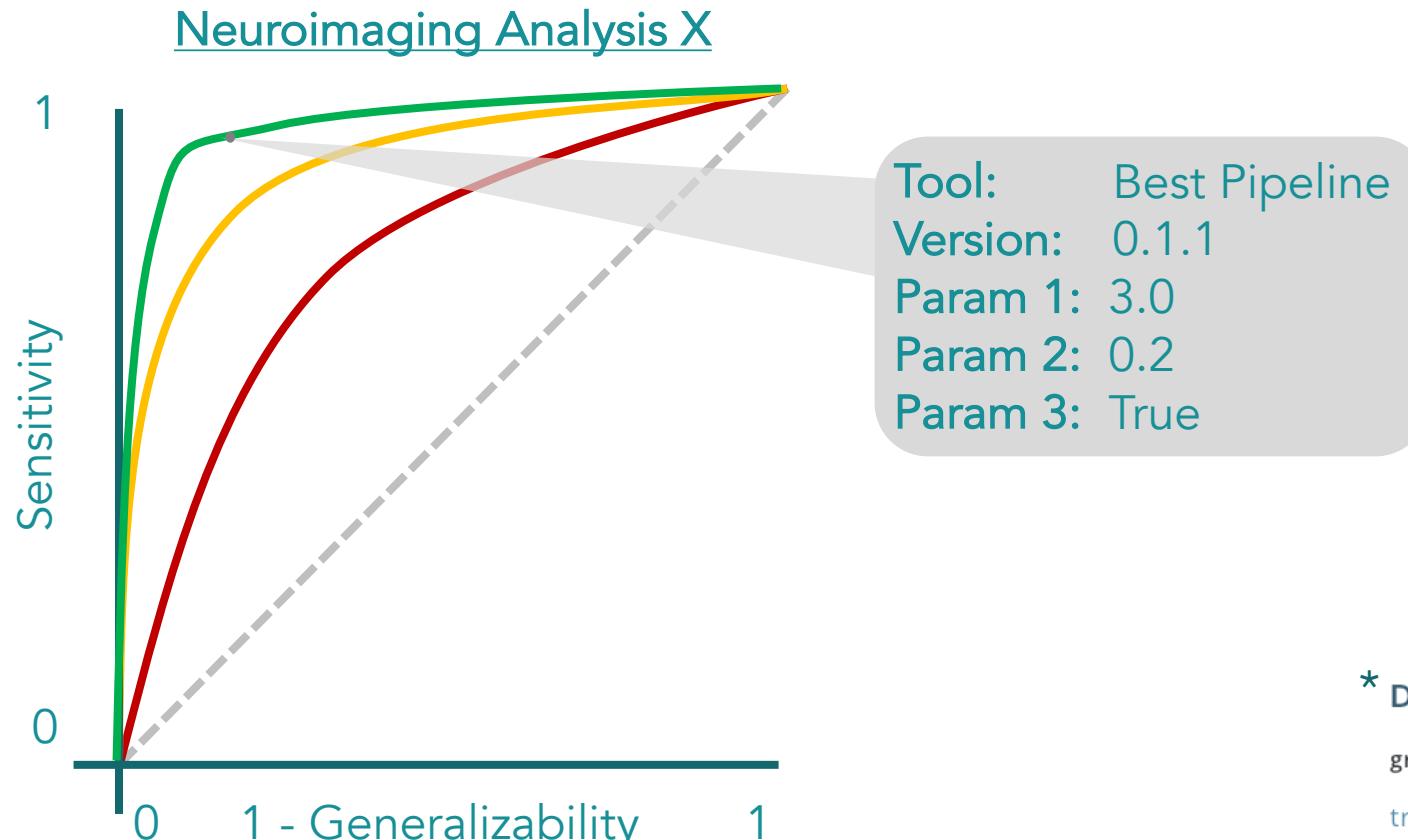
“the likelihood that models of similar groups yield similar results”



Limitations of these works

- Discriminability does not consider the quality of the data
- NPAIRS relates correlations of derivatives, limiting data type
- NPAIRS models linear differences (when inference may be non-linear)
- Neither characterize the variance of differences between groups
- Both compare *data*, rather than *claims made from the data*

Ideally, we'd have Generalized-ROCs (GROCs)*



* Definition of GROK

grokked; grokking

transitive verb

: to understand profoundly and intuitively

The plan is...

1. Detailed literature review to uncover models of generalizability
 - Potentially turning this into a review paper on pipeline evaluation
2. Deploy BIDS apps on heterogeneous dataset
 - Freesurfer (structural MRI), ndmg (diffusion MRI), C-PAC (functional MRI)
 - Consortium of Reliability and Reproducibility (CoRR)
3. Model the sensitivity and generalizability of each tool using various operating points/parameter selections and generating GOCs

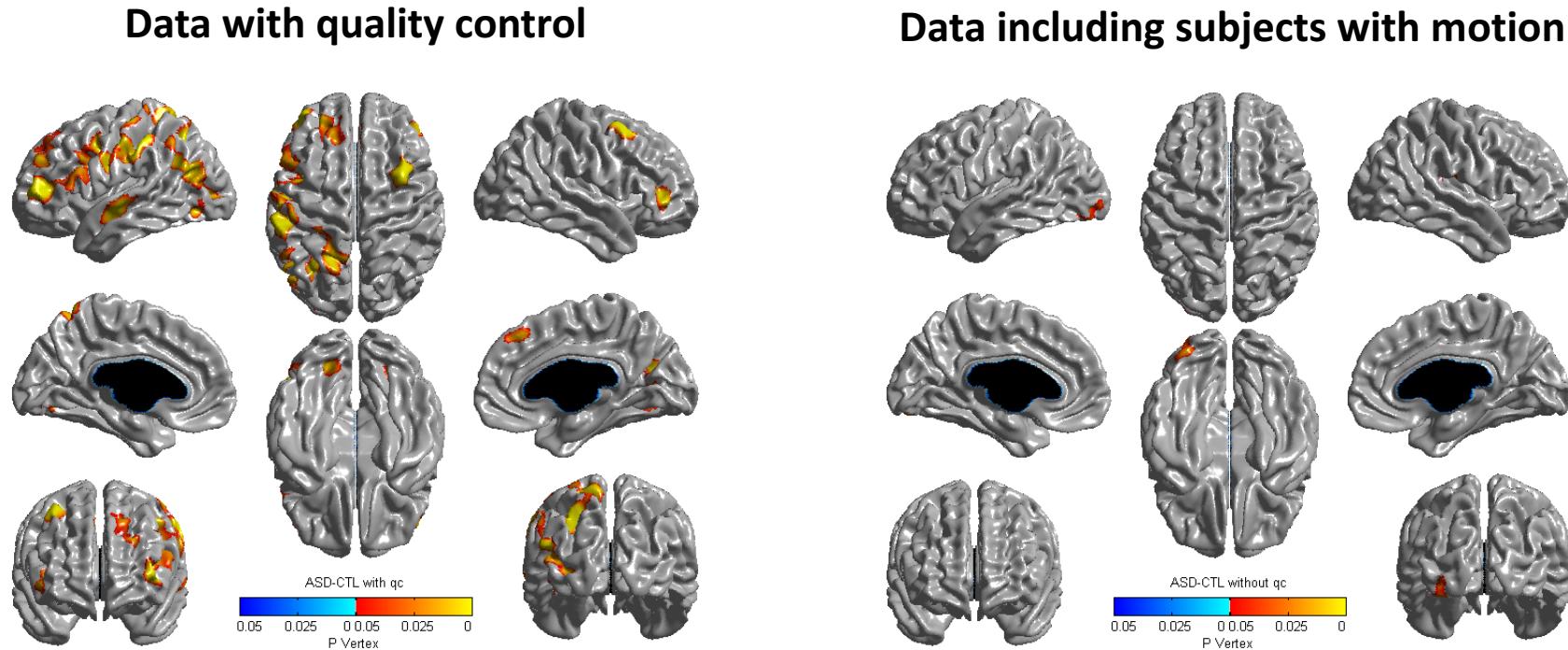
Chapter 3: Pipeline optimization in varying inference contexts

1 year (total: 3.5 years)

The issue is...

- Datasets are different...
 - participant demographics,
 - scanner manufacturers,
 - number of participants,
 - similarity of acquisition protocols, etc.
- Tools perform differently in different contexts
- There's no a priori method for determining processing choices given knowledge about a dataset

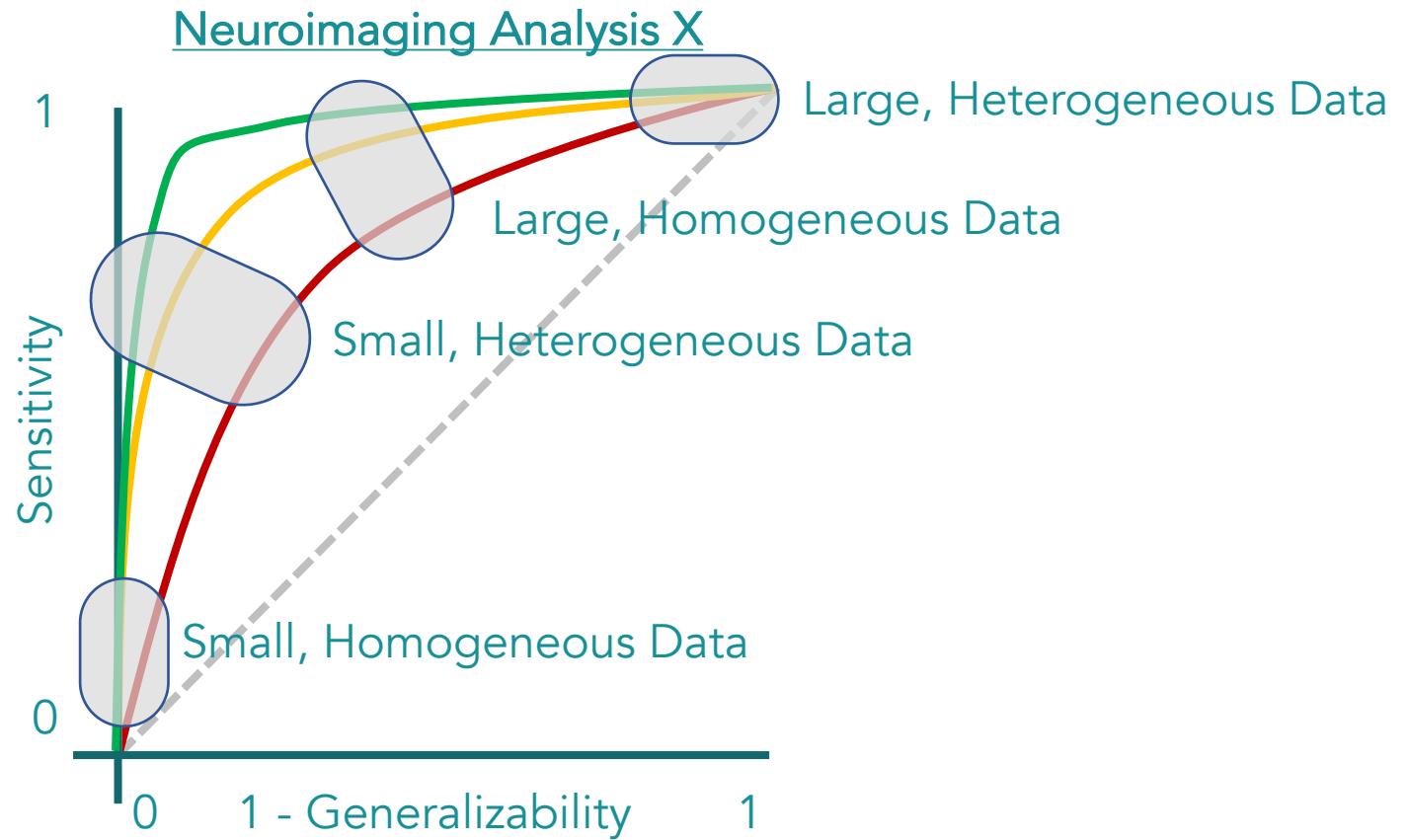
Differences in ABIDE nulled with motion



Robustness of processing and strictness of quality control can meaningfully change resulting scientific claims (Khundrakpam et al., 2017)

Identify regions of interest on the GROCs

Based on the dataset, where on the GROC do I want to perform my experiment?



The plan is...

1. Process data from different “contexts” with the previously-characterized tools, using settings from various points on the GROC

Size	Homogeneous	Example Dataset
Small	No	Subset of CoRR
Small	Yes	Kirby21
Large	No	ABIDE
Large	Yes	HCP

2. Determine optimal regions of interest on GOCs for each context

Conclusion

A common worst-case approach to neuroscience

1. Pose a hypothesis
 2. Collect + curate dataset
 3. Manually perform QC on dataset
 4. Pick processing pipeline and parameters
 5. Process random subset with pipeline in 4.
 6. Manually perform QC on derivatives
 7. Redo from 4. if not happy with 6.
 8. Process all data with pipeline in 4.
 9. Answer statistical question
 10. Publish claim
- Skipped or expedited
← (Khundrakpam et al., 2017) vs. (Haar et al., 2016)
- Arbitrarily chosen (i.e. defaults)
← Homogeneous sub-sample
- Prone to overfitting, undocumented
← Heterogeneous collection, no provenance
- Of questionable reproducibility
(Open Science Collaboration, 2015)

A better ~~worst-case~~ approach to neuroscience

1. Pose a hypothesis
 2. Collect + curate dataset
 3. Manually perform QC on dataset
 4. Pick processing pipeline and parameters
 5. Process random subset with pipeline in 4.
 6. Manually perform QC on derivatives
 7. Redo from 4. if not happy with 6.
 8. Process all data with pipeline in 4.
 9. Answer statistical question
 10. Publish claim
- Potentially mitigated through subsequent tool selection ← Contextually chosen ← Including external data ← Reduced overfitting, provenance captured ← Increased portability, provenance captured ← That is now more reproducible and generalizable

<Demos />

Clowdr – prototype, launch, and share an analysis

Apine – query example BIDS datasets

Acknowledgements

- McGill Centre for Integrative Neuroscience (Alan Evans, Shawn Brown, et al.)
- Big-Data for Neuroinformatics Lab (Tristan Glatard, et al.)
- Jean-Baptiste Poline, Pierre Bellec, Christine Tardif
- Montreal Neurological Institute/The Neuro
- Healthy Brains for Healthy Lives
- Open Science Beers
- Lab-mates, Friends, Family, Community, Universe

All code mentioned in this presentation is publicly available on GitHub.

Thanks!

Find me @



gkiar



g_kiar



greg.kiar@mcgill.ca



Integrating the above (i.e. “the dream”)

- Go to a website
- Pick a dataset
- Pick an analysis
- Design a hypothesis
- Launch it
- Go outside & run around
- Come back to your answer
- Share the results, form new hypotheses, and collect new data

