

---

# **Food Regulation fMRI Documentation**

*Release 1.0*

**Daniel J Wilson**

**Aug 31, 2020**



---

## Contents:

---

<b>1</b>	<b>Links</b>	<b>1</b>
1.1	Getting Started . . . . .	1
1.1.1	Running the Experiment . . . . .	1
1.1.2	Data Storage . . . . .	1
1.1.3	Updating Documentation . . . . .	2
1.2	Folder Structure . . . . .	2
1.2.1	1_project_management . . . . .	2
1.2.2	2_ethics_governence . . . . .	3
1.2.3	3_experiment . . . . .	3
1.2.4	4_dissemination . . . . .	3
1.2.5	docs . . . . .	3
1.3	Experiment Details . . . . .	4
1.3.1	Protocols . . . . .	4
1.3.2	Main Task . . . . .	4
1.3.3	Localizers . . . . .	5
1.3.4	Questionnaires . . . . .	5
1.4	Data Analysis . . . . .	6
1.4.1	fMRI Preprocessing . . . . .	6
1.4.2	GLMs . . . . .	7
1.4.3	Behavioral . . . . .	7
1.4.4	Neural . . . . .	7
1.4.5	Correlations . . . . .	8



- Source code: [https://github.com/danieljwilson/cogReg\\_fmRI](https://github.com/danieljwilson/cogReg_fmRI)
- Contact: [daniel.j.wilson@gmail.com](mailto:daniel.j.wilson@gmail.com)

## 1.1 Getting Started

This documentation will cover some of the basics of the project, and hopefully put you in a position to run the experiment and analyze data yourself.

### 1.1.1 Running the Experiment

The experiment protocol can be found on Dropbox at `DJW_Projects/02_FOOD_REG/PAPERWORK/fMRI_Food_Regulation_Experiment_Protocol.gdoc`

The protocol includes information regarding:

- Booking Scanner Time
- Recruiting Participants
- Running Study

The study questionnaire is a Google Form, and lives on the lab's Google Drive account.

### 1.1.2 Data Storage

All data (raw and processed) are stored on external hard drive `CH_ext_001`. All files are in the folder `2019_FoodReg_fmRI/`.

Note that this drive is password protected.

### 1.1.3 Updating Documentation

**Note:** Keep in mind that if you are using the Read the Docs documentation there is always the option to add/edit.

---

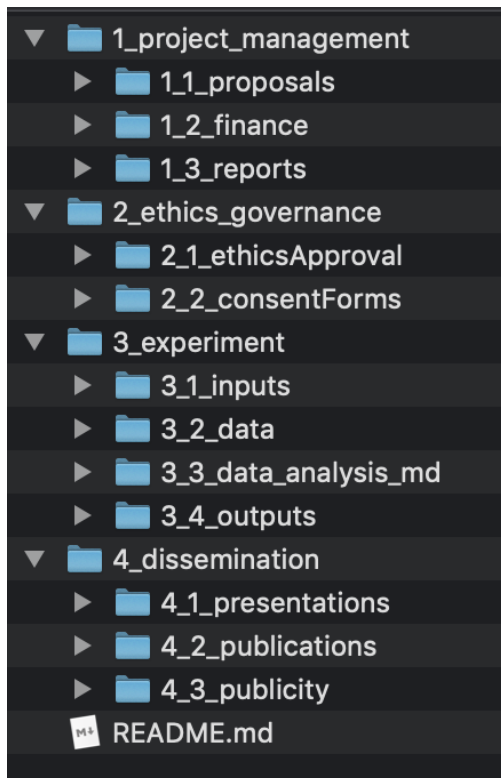
Just look for this image on the top right of the page:



Click on it (which automatically forks it), make your edits and then create a pull request.

## 1.2 Folder Structure

The folder structure for the project follows the format illustrated below:



---

### 1.2.1 1\_project\_management

This is not uploaded to git.

## 1.2.2 2\_ethics\_governance

This is not uploaded to git.

## 1.2.3 3\_experiment

This is where most of the project lives.

### 3\_1\_inputs

Refers to the tools used to capture information, including:

- **Experiment code**
  - including all assets (e.g. photos)
- Questionnaires

### 3\_2\_data

Raw data lives here.

### 3\_3\_data\_analysis

This includes:

- Scripts for preprocessing and cleaning data
- Processed data
- Scripts for analyzing processed data

### 3\_4\_outputs

This includes:

- Plots
- Tables
- Markdown results section (theoretically)

## 1.2.4 4\_dissemination

Presentations, publications and publicity live here.

## 1.2.5 docs

There is also a folder that has been added by [Sphinx](#), and is where all the documentation lives.

---

**Note:** Keep in mind that if you are using the Read the Docs documentation there is always the option to add/edit.

---

Just look for this image on the top right of the page:



Click on it (which automatically forks it), make your edits and then create a pull request.

## 1.3 Experiment Details

### 1.3.1 Protocols

The experiment protocol is located on Dropbox at: `DJW_Projects/02_FOOD_REG/PAPERWORK/fMRI_Food_Regulation_Experiment_Protocol.gdoc`

The fMRI scan protocol is [here](#).

### 1.3.2 Main Task

The main task involved three discreet phases.

1. Pre-Scan
  - Food liking ratings
  - Main task training
2. Scan
  - In-scanner trials
    - 9 runs
3. Post-Scan
  - Food liking ratings (repeated)
  - Food taste ratings
  - Food health ratings

---

The code for the Main Task (including the instructions) was written in MATLAB and uses [Psychtoolbox](#).

The [Main Task Code](#) includes many files, but the key scripts are:

- `runPreMRI.m`
  - Launches the experiment instructions and the initial Food Liking Rating
- `runSession.m`
  - Launches a run of the experiment in-scanner
- `runPostMRI.m`
  - Launches the post task ratings of Liking, Taste, and Health



### 1.3.3 Localizers

The localizer task involved two discrete phases.

1. Pre-Scan
  - Localizer task training
2. Scan
  - In-scanner trials
    - go-nogo: 2 runs
    - switching task: 1 run

The code for the Localizers was written in Python, using the Psychopy toolbox.

#### Go-NoGo

The [Go-NoGo task scripts](#) include both a `practice` and `fMRI` version. The main difference is that the `fMRI` version waits for the scanner to send a `5` to progress.

The Go-NoGo task was based on [Wager et al. 2005](#). We used the letters ‘m’ and ‘w’ as the ‘go’ and ‘no-go’ stimuli, requiring the execution or withholding, respectively, of a keypress response (counterbalanced).

After presentation of a 500ms fixation cross, participants had 450ms to respond to the stimulus.

There were two types of blocks presented: low-go blocks, in which 20% of the trials required a response, and high-go blocks, in which 50% of trials required a response. The beginning and ends of these blocks were not indicated to participants.

A total of 24 blocks (12 of each condition), containing 12 trials each—a total of 288 trials—were presented. The rapid event-related design with clustered events is expected to maximize power ([Liu 2004](#)). The task was broken into two equal (144 trial) sessions for scanning to reduce fatigue.

#### Switching Task

The [switching task scripts](#) also include both a `practice` and `fMRI` version. The main difference is that the `fMRI` version waits for the scanner to send a `5` to progress.

The attention switching task showed subjects a pair of images, one face and one house, on each trial. The images were overlaid directly on top of each other with each image’s opacity reduced so that both images could be clearly deciphered. On each trial subjects were directed to focus their attention either on the Face or the House image, which was indicated both by text on screen indicating “Face” or “House” and the background color of the image (i.e. a different background color for Faces and Houses). On Face trials, subjects had to determine the face’s gender, using a keypress to indicate their response. On House trials, subjects had to indicate whether it was an old or modern house, using a keypress to indicate their response.

There were four total response possibilities, and four corresponding buttons to press. Participants had up to 1 second to respond. The inter trial interval was between 1s and 6s, uniformly distributed. A total of 80 trials were presented in a single session.

### 1.3.4 Questionnaires

Upon completion of all tasks we asked subjects to complete the following questionnaires:

1. [Three-Factor Eating Questionnaire](#)

2. Rapid Food Screener
3. Barrat BIS 11
4. Perceived Stress Scale

There is also a questionnaire key, that will be helpful for data analysis.

## 1.4 Data Analysis

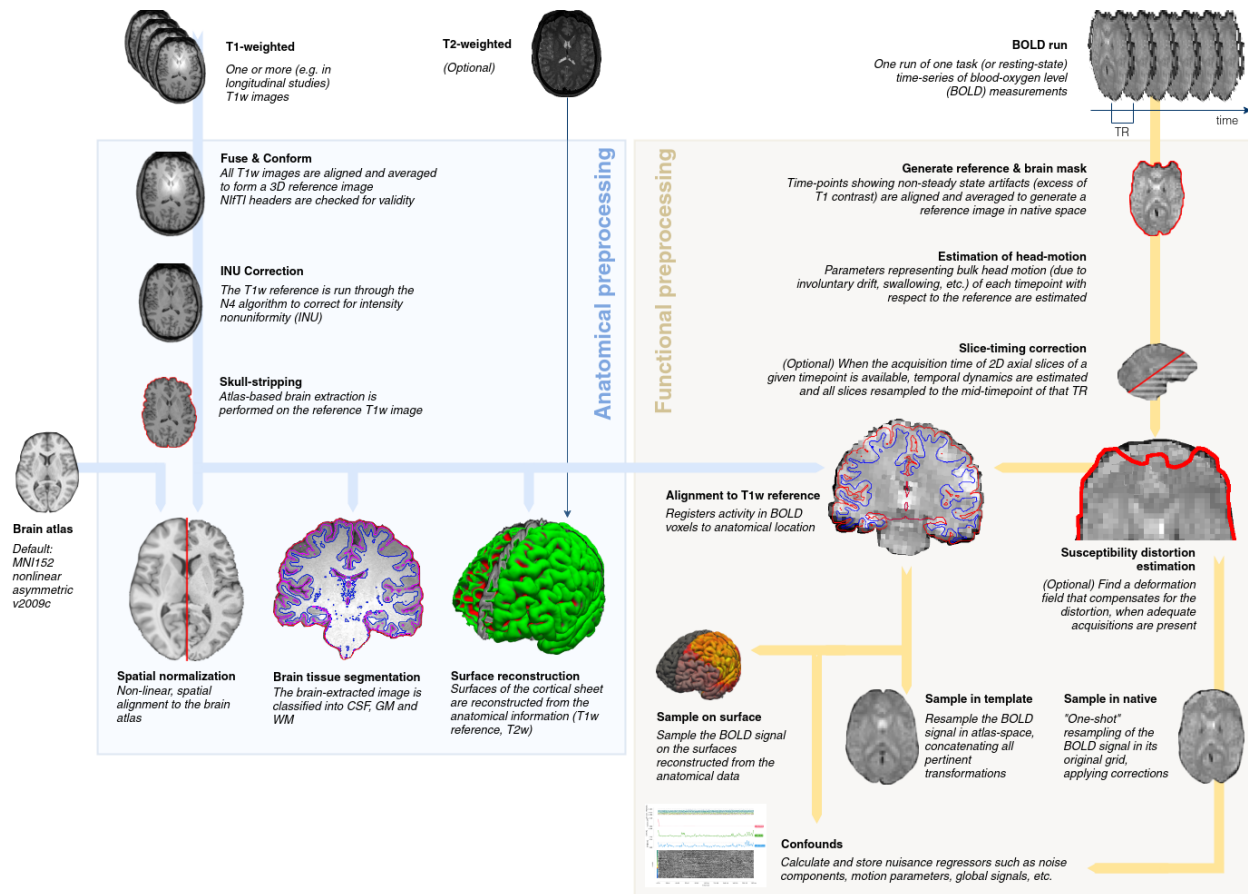
### 1.4.1 fMRI Preprocessing

#### fmriprep

`fmriprep` is a pipeline developed by the Poldrack lab at Stanford University for use at the Center for Reproducible Neuroscience (CRN), as well as for open-source software distribution.

`fmriprep` is designed to provide an easily accessible, state-of-the-art interface that is robust to variations in scan acquisition protocols and that requires minimal user input, while providing easily interpretable and comprehensive error and output reporting.

It performs basic processing steps (coregistration, normalization, unwarping, noise component extraction, segmentation, skullstripping etc.) providing outputs that can be easily submitted to a variety of group level analyses, including task-based or resting-state fMRI, graph theory measures, surface or volume-based statistics, etc.



The `fmrip` workflow takes as principal input the path of the dataset that is to be processed. The input dataset is required to be in valid BIDS (Brain Imaging Data Structure) format, and it must include at least one T1w structural image and (unless disabled with a flag) a BOLD series. We highly recommend that you validate your dataset with the free, online [BIDS Validator](#).

The exact command to run `fmrip` depends on the Installation method. The common parts of the command follow the [BIDS-Apps](#) definition. Example:

```
fmrip data/bids_root/ out/ participant -w work/
```

## Running on a Cluster

In order save time it makes sense to do preprocessing on a cluster (like [SciNet](#)). To do so you will need an account.

More information can be found [here](#) about options.

### 1.4.2 GLMs

General linear model scripts were run using MATLAB and SPM 8.

The model regressors are specified by the files ending in `analyze2`.

The contrasts are calculated in the files that start with `contrast2`.

The second level/group analyses are performed by the `rfx_par` script.

While the `analyze` and `contrast` scripts can be run just with the function, you need to use the following syntax to run the `rfx_par` script. Note that you need to provide access to a contrast file.

Example:

```
f = fullfile('8_pre_liking', preproc_version, 'm8_pre_liking_cons.mat');
load(f);
for con = 1:length(cname)
    rfx_par('8_pre_liking', cname(con), good_subjects, preproc_version)
end
```

### 1.4.3 Behavioral

A Jupyter notebook (using an R kernel) for the [behavioral results](#).

#### DDM

We fit both a base model and constant model (with an additional *constant* parameter that is added to the drift).

The [DDM model scripts](#).

A Jupyter notebook (using an R kernel) for the [ddm results](#).

### 1.4.4 Neural

A Jupyter notebook (using an R kernel) for the [neural results](#).

## 1.4.5 Correlations

TO BE ADDED...