preRNG verification

Bob

Here I validated the pre-registration of "*Cerebellum Involvement in Hand Movements: a Functional MRI Study*" (Alice, 2017). Although Alice's paper is focused on the cerebellum, I decided to use primary cortical activations as a voucher for the pre-registration validity. I found these activations to be aligned with the randomization induced by the protocol folder, verifying the study's pre-registration.

1. I ran the function `createEVs` (part of `analyze.py` that is found in the protocol folder) specifying the path to the compressed protocol folder as argument. This function calls the `preRNG.py` function, which initialized PRNG to the hashed protocol sum. It then generated a pseudorandom order of blocks for each of the four runs. This resulted in the following protocol sum and order of blocks:

   ```
   >>> createEVs(os.path.join(home_dir,'protocolFolder.zip'))
   '620d185c26237de2e54a0affaeb7a9d2b9ca5b0185ec1f1e90796fe6ff06152e'
   ```

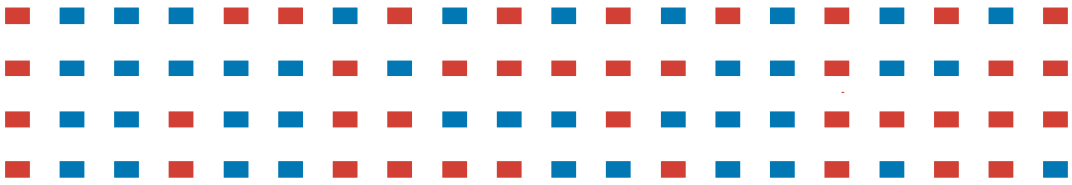

Figure 1: The resulting order of blocks in the four experimental runs. Blue rectangles represent blocks in which the subject was requested to move her right hand, and red rectangles represent blocks in which she was requested to move her left hand.

2. I performed a whole-brain contrast between right and left hand movements using the information I acquired about the block order in the previous step:

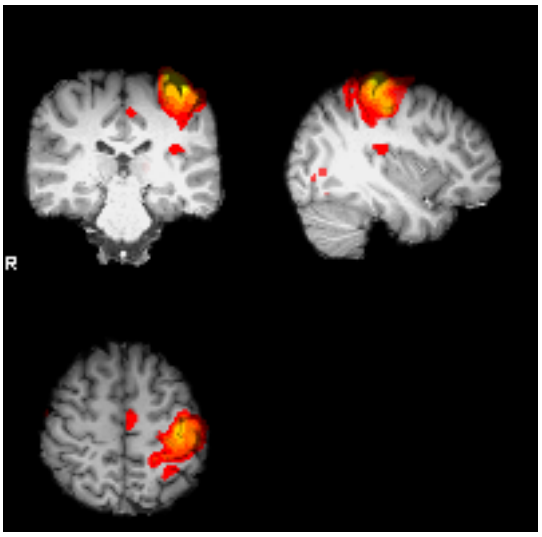

Figure 2: a contrast of right hand > left hand. Results are corrected for multiple comparisons, whole brain.

3. Introducing slight changes to the protocol folder (by adding a sequence of # symbols to the end of the analysis file found in the protocol folder) resulted in activations weaker by orders of magnitude:

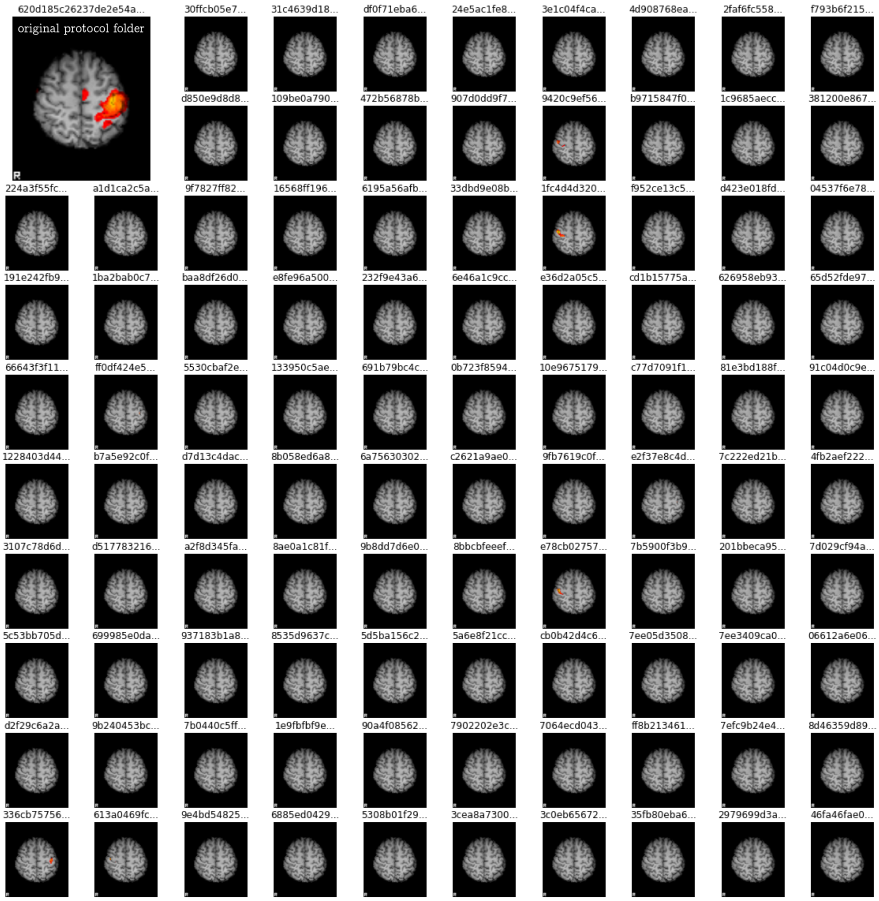

Figure 3: Appending #s to the file titled analyze.py and applying the preRNG function to the altered protocol folder dramatically changed the protocol-sum (first 10 letters appear above the image) and the presumed order of events. This slight change resulted in activation maps weaker by orders of magnitude compared to the one obtained from from the original protocol folder.

4. I extracted activation timecourses from two voxels in the right and left central sulci. These timecourses aligned well with the order of events as obtained from applying the preRNG function to Alice's original protocol folder:

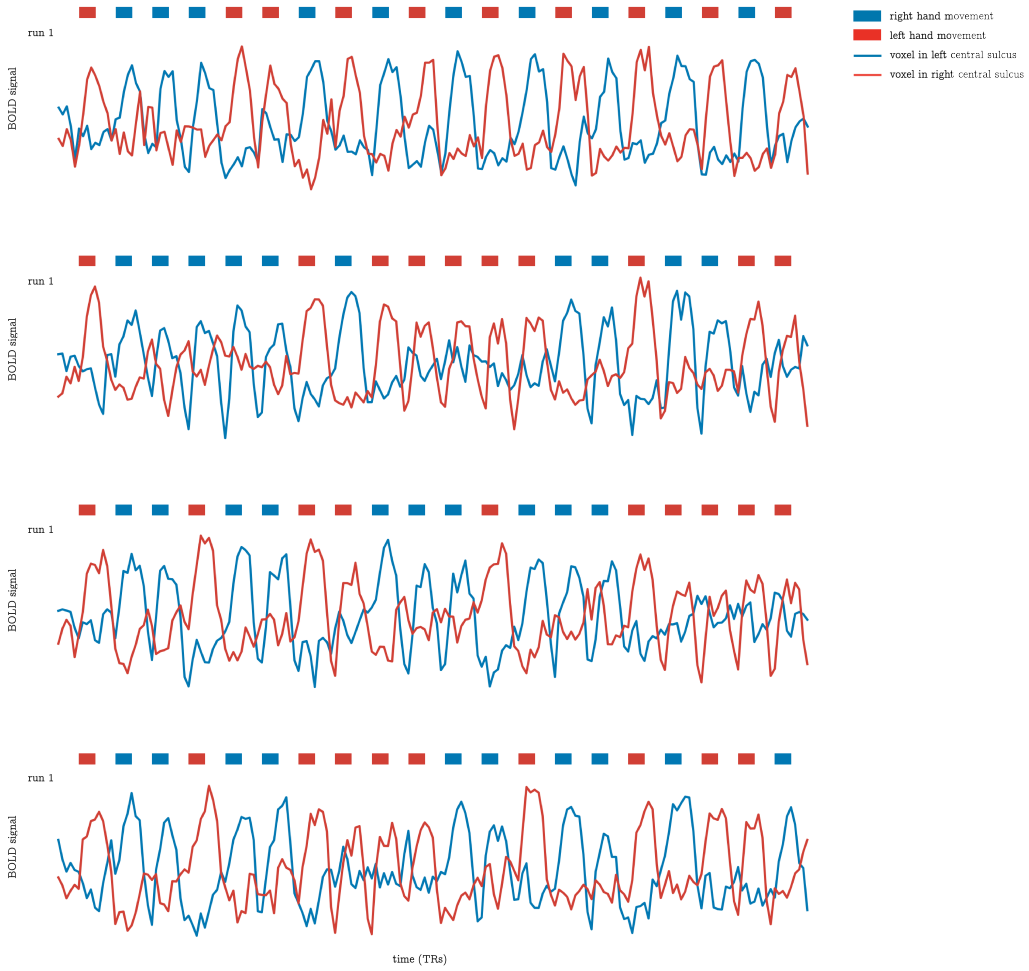

Figure 4: Activation timecourses for the four experimenta runs, as extracted from the bilateral hand knobs in the central sulci. blue rectangles represent blocks in which the subject was requested to move her right hand, and red rectangles represent blocks in which she was requested to move her left hand. Blue and red lines correspond to the left and right central sulci accordingly.

5. To further explore the alignment of the primary cortical activations with the presumed order of events, I computed the ratio between the likelihood of the voxels' timecourses given neighboring experimental designs (designs that are identical to the original, except for one swap of right and left experimental blocks; $Design'$) and the original experimental design ($Design$). I computed this Bayes factor ($\frac{p(S1activations|Design'}{p(S1activations|Design}$) for each neighboring experimental design, using a naive linear regression model (not accounting for autocorrelations). None of the Bayes factors exceeded 1, suggesting that the original experimental design maximized the likelihood of these data, at least locally.

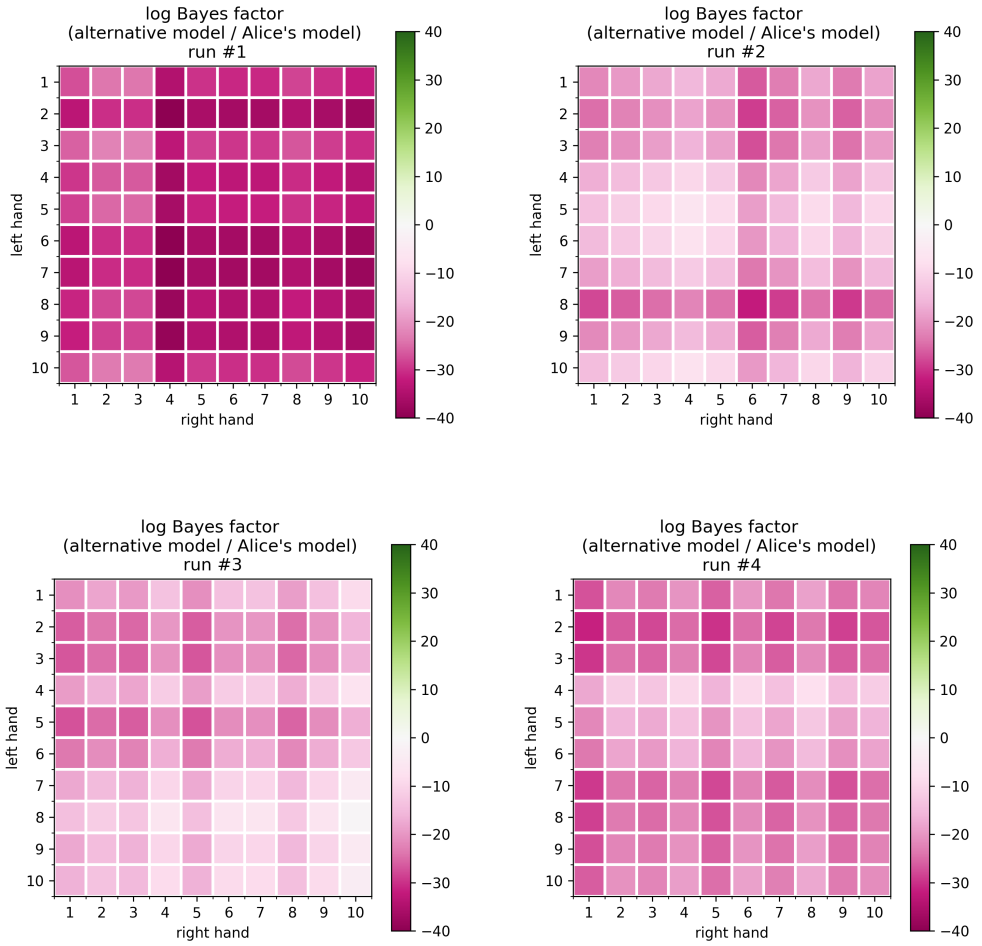

Figure 5: Matrix entries represent the log likelihood ratio between the neighboring design and the original order of events. For example, entry 9,4 in the first matrix compares the likelihood of the data given a design in which the ninth left hand event has been replaced with the fourth right hand event, to the likelihood of the data given the original experimental design. The log-likelihood is negative, suggesting that the alternative model is less appropriate. Note that none of the swaps across all four experimental runs increases the likelihood of the data.

6. Finally, to examine whether other event orders maximize the likelihood of the chosen voxels' activations, I performed gradient descent optimization starting from an arbitrary order of events. I then performed the following steps:

   (a) Generate the log Bayes factor matrix A by computing a Bayes factor for each possible swap of left and right events, as in step 5

   (b) Extract the maximum log Bayes factor $A_{i,j}$ from the matrix

   (c) if $A_{i,j} > 0$, swap the i^{th} left hand event with the j^{th} right hand event, and repeat steps a-c. otherwise, stop.

   For all four runs, the algorithm converged to the same order of events that was obtained by using the preRNG function with Alice's protocol folder, validating the alignment of this randomization with the raw experimental data.