Montijn’s Merry Matlab Manual

There are many functions in Processing/analysis/ that are either obsolete or irrelevant, so I will only discuss a subset of them. If the specific function is particularly important, I’ve put it in bold (i.e., #3 and #8).

**General overview**

runModelAnalysis[\*].m

runSimAnal[\*].m

Old analysis scripts not used anymore

runSimAnalBlock[\*].m

These are the actual analysis scripts I use to make the graphs. See below for more details.

runMetaBlock[\*].m

These produce aggregate graphs, based on data files produced by the corresponding runSimAnalBlack function.

**Specific functions**

runSimAnalBlock0

Runs Fourier analysis, mostly to check for non-stationarities in data

runSimAnalBlock1

Runs Fisher information analysis, as a function of number of neurons

runSimAnalBlock2

Runs spiking coincidence analysis. Not used recently

runSimAnalBlock3

Produces some graphs to show model responses (connectivity, receptive fields, etc)

**runSimAnalBlock4**

Subspace analysis with Fisher information per dimension of the subspace. This is one function you’re likely to use a lot.

runSimAnalBlock5

Same as #4, but compares what happens if the subspaces are calculated on the combined noise subspaces from neighbouring stimuli (unsplit) or separately for the two stimuli (split).

runSimAnalBlock6

Same as #4, but I used this to look at different population sizes.

runSimAnalBlock7

Same as #4, but for merging the two sets (i.e., pretending they’re from the same stimulus) into a single covariance matric, thereby artificially inducing post-hoc differential correlations.

**runSimAnalBlock8**

Analysis for cos^2(f’,\Sigma) as a function the eigen-subspace of \Sigma. The function has some variants for looking at the contributions of particular neurons.

runSimAnalBlock9

Similar to #8, but now for Fisher information as a function the eigen-subspace of \Sigma. The function has some variants for looking at the contributions of particular neurons.

runSimAnalBlock10

Runs many bootstraps on the data to get a distribution of \phi values that can be expected, given the number of neurons and trials in the experimental data set, had the differential correlations been the strength that we put into the simulation.

runSimAnalBlock11

Fisher information as a function of number of neurons. Difference with #1 is that this function is made to produce results comparable to #8 and #9: steps of single neurons, rather than the larger steps as in #1, and the graphs are normalized between 0 and 1.

runSimAnalBlock12

Analyzes the principal components of \Sigma, and produces various graphs, such as \lambda at the i’th component of \Sigma, for a range of population sizes.

**Final note**

#4, and #7-#11 produce data files that their respective Meta functions use to produce aggregate graphs.