Readme for behavioral data and Matlab tutorial accompanying *Dynamic weighting of multisensory stimuli shapes decision-making in rodents and humans*.

Written by John Sheppard, August 4, 2013.
Questions? sheppard@cshl.edu

Raw behavioral data

Behavioral data files are saved in the data folder as Matlab *.mat* files and can be imported directly into Matlab. Mat files contain a data structure, *dat*, that is loaded into the workspace upon importing the file into Matlab. Each index of the data structure pertains to an individual trial, ordered consecutively based upon their presentation to the subject within an individual behavioral session. Variable names and conventions were preserved between the rat and human datasets where possible (see below).

Structure fields common to rat and human datasets

sessionDate: Date of behavioral session (dd-mmm-yyyy)

sessionNumber: Unique numeric identifier of the specific behavioral session from which trial was obtained for the given subject

aud/visStimPresented: 0/1 specifies absence/presence of aud/vis stimulus on specified trial

aud/visEventIntervals: Vectors of 1's and 2's indicating sequence of short/long inter-event-intervals presented on the trial. Vectors are empty if no aud/vis stimulus was presented.

aud/visEventTimes: Vectors indicating millisecond times of individual event arrivals for the aud/vis stimuli, timed relative to stimulus onset (e.g., 1 = first millisecond of stimulus).

aud/visEventRate: Trial-average rate of aud/vis events presented on specified trial (NaN indicates no stimulus was presented). Because each stimulus was 1 second in duration, this is equivalently the number of auditory or visual events presented on the given trial.

aud/visRelLevel: 1/2 specifies low/high reliability trial for corresponding modality (NaN indicates no stimulus was presented). For rats, only a single visual reliability level was presented.

audEventLoudness: Loudness of auditory stimulus (dB SPL); NaN: no stimulus presented

audNoiseLoudness: Loudness of auditory background noise (dB SPL); NaN: no stimulus presented

audSNRdB: SNR of auditory stimulus (Event dB SPL - Noise dB SPL)

visEventLuminance: Luminance of visual stimulus events (units of candela/m^2)

cueConflict: Cue conflict of trial (cueConflict = Visual event rate - auditory event rate; NaNs: single sensory trials)

outcome: 0/1 indicates punishment/reward signal for both species; -1 indicates no feedback given (human); -2 indicates early withdrawal punishment (rat); -3 indicates the rat did not respond within the allotted time after stimulus offset (triggering an additional 2-second timeout punishment).

response: 1 and 2 indicate low-/high-rate decision for either species (i.e., left/rightward movement or keypress); 999 indicates an invalid keypress (other than left/right) for humans or an aborted trial for rats.

trialDuration: Duration of trial (ms). This value is fixed at 1000 ms for both species.

eventDuration: Duration (ms) of individual stimulus events. This value is fixed at 10 ms (humans) or 15 ms (rats).

intervalDurations: Durations (ms) of short/long inter-event intervals used for aud/vis stimuli. (This does NOT include the duration of stimulus events. That is, the time between the onset of two consecutive events equals the inter-event-interval PLUS the duration of the first event; see 'eventDuration' above.)

Structure fields specific to human data

visSNRdB: SNR of visual stimulus, computed as 10*log10(Event luminance / Noise luminance)

visSNRRatio: SNR of visual stimulus, computed as Event luminance / Noise luminance.

Structure fields specific to rat data

imposedWaitDuration: Required wait duration of rats; wait durations below this value result in "early withdrawal" punishments and are considered aborted trials.

actualWaitDuration: Actual duration (milliseconds) rat occupied center port from beginning of stimulus ("Center In") until withdrawal ("Center Out")

PsychFits data structures

Also included in the data folder are "psychFits" data structures. These mat-files contain psychometric measures computed from each subject's raw data that are used to generate the plots in Figures 2-6. They have been pre-compiled from the raw data for convenience using the included script, *GetPsychometricFits.m*. This script calls various analysis functions included with this tutorial to compute the psychometric fits. Note that in order to fit psychometric functions to subject's choice data (Figs. 2-3), the psignifit toolbox for Matlab must be installed and working. This toolbox is available at http://psignifit.sourceforge.net. However, because the fit psychometric functions and other measures have been pre-compiled, it is not necessary to install this toolbox to plot the psychometric data in Matlab.

Description of structure fields within *psychFits* data structures

*trialTypes**: the highest-level subfield separates all psychometric data by trial type

- *name:* string identifier of trial type, e.g. 'audLowRel_visHighrel'

- *choiceDatabyRate:* nx3 summary array of subject responses organized by trial event rate. Each row specifies: [Trial event rate, # high-rate responses, total # responses]

- *psychParams:* subfield containing estimates and standard errors of parameters for the psychometric function fit to the corresponding trial type

  -- *name:* String identifier of psychometric function parameter (see FitPsychometricCurve.m for details)

  -- *est, stdErr*: Estimate and standard error for corresponding parameter obtained from psignifit via Monte Carlo resampling of the choice data

- *psychCurve*: nx2 array containing discretized approximation of continuous psychometric curve fit to the choice data.
  Column 1: Vector of event rates (events/second)
  Column 2: Vector of choice probabilities (P(high-rate decision))

* Sub-fields specific to single sensory trial types:

  **excessRateResults*: Results of excess rate analysis returned by *ComputeExcessRate.m*

  -- *stimDuration*: Specified stimulus duration (ms) over which excess rate curve is computed
  -- *binWidth*: Specified width of sliding window used to compute excess rates.
  -- *timeVector*: Vector of stimulus times corresponding to the centers of the sliding windows at each position excess rate is computed within trial duration
  -- *leftChoiceMeanWindowRates*: Mean event rates within the sliding windows for selected trials that preceded leftward (low-rate) decisions
  -- *rightChoiceMeanWindowRates*: Mean event rates within the sliding windows for selected trials that preceded rightward (high-rate) decisions
  -- *excessRateEsts*: Estimates of the excess rate at each position of sliding window (equal to *rightChoiceMeanWindowRates - leftChoiceMeanWindowRates)*
  -- *excessRateStdErrs*: Standard errors of the excess rate estimates, determined via bootstrap analysis
  -- *shuffledExcessRateEsts*: Estimates of the excess rate after assigning random choices to each selected trial that are sampled randomly from the dataset's choice distribution (used as a control to assess significance of the excess rate)
  -- *shuffledExcessRateStdErr:* Standard errors computed for the shuffled excess rates.

**Sub-fields specific to multisensory trial types:

**_conflictLevels:_ Separates psychometric data by cue conflict level
-- _cueConflict_: Visual event rate - auditory event rate

**_predVisWeightEst/StdErr_: Estimate/standard error of predicted visual weight

**_visWeightEst/StdErr_: Estimate/standard error of observed visual weight

Plot generation scripts

The included Matlab scripts (plot_figureX.m) can be used to reconstruct Figures 2-6 within Matlab. See comments in individual m-files for details regarding how each individual plot is constructed.

Analysis functions

See comments to code within the individual m-files for further details on these functions.

* Requires psignifit toolbox for MATLAB.
** Requires MATLAB Symbolic Math Toolbox.

ComputeExcessRate.m
Calculates the “excess rate” (reverse correlation) curves presented in Figs. 5-6.

simulateEvidenceAccumulation.m
Computes simulated “excess rate” curves resulting from a decision-maker that accumulates evidence over arbitrary periods of the trial duration (Figs. 5a-b).

movingAverage.m
Computes moving averages on time series data and is used to smooth out event-related artifacts when plotting the “excess rate” curves. This function was written by Carlos Adrián Vargas Aguilera and is also available online at the MATLAB Central File Exchange: http://www.mathworks.com/matlabcentral/fileexchange/40758-filter-smooth-calculating-the-moving-average-along-a-vector/content/moving_average.m

boundedLine.m
Plots a line surrounded by shaded confidence bounds. This function is used to plot the excess rate estimates and standard error curves in Figs. 5-6. The function was written by Kelley Kearney (2010-2013) and is freely available online at the MATLAB Central File Exchange: http://www.mathworks.com/matlabcentral/fileexchange/27485-boundedline-line-plots-with-shaded-errorconfidence-intervals

*FitPsychometricCurve.m**
Fits a cumulative Gaussian psychometric function to behavioral choice data by calling functions from the psignifit toolbox for Matlab.

*ComputeVisualWeight.m***
Calculates observed visual cue weights and standard errors from subject's multisensory performance.

*PredictVisualWeight.m***
Calculates predicted visual cue weights and standard errors given the subject's single sensory performance.

*PropError.m***
Computes estimates and standard errors for any derived quantity that is a function of variables with defined estimates and standard errors. The standard error is calculated via propagation of uncertainty. This function is used to compute standard errors for the observed and predicted visual cue weights reported in the paper. The function was written by Brad Ridder (2007) and the version included with this tutorial was slightly expanded from its original version following the BSD guidelines. The original function is freely available online at the MATLAB Central File Exchange:
http://www.mathworks.com/matlabcentral/fileexchange/17901-propagation-of-uncertainty/content/PropError.m