

Code and documentation by Yoram Ben-Shaul
yoramb@ekmd.huji.ac.il

Neuronal response simulator

Overview

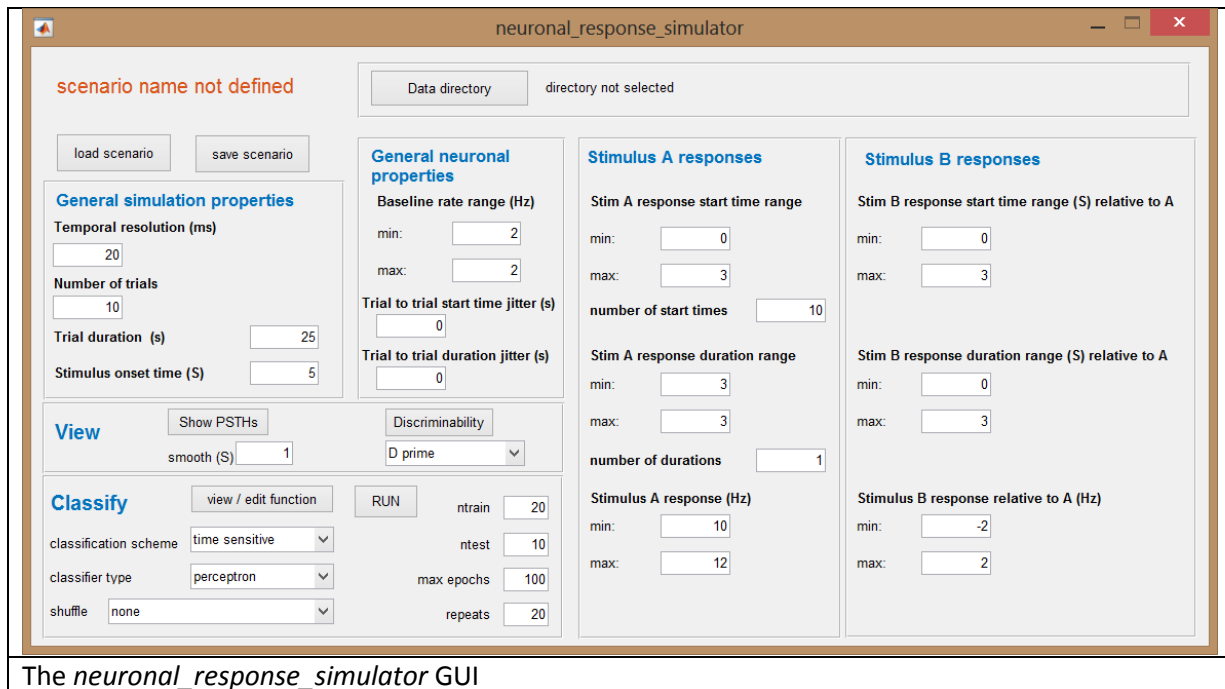
The goal of the Neuronal response simulator is to provide insights about how response features (with a focus on temporal parameters), affect the ability to determine stimulus identity from neuronal response. The simulator allows specification of responses of a neuronal population to two different stimuli (stimulus A and B). Responses of each neuron can differ in their time course, and in their magnitudes. Once a population is defined, it is possible to visualize the PSTHs, view the discriminability in responses, and to conduct classification analysis on the dataset. The program also creates data files with the simulated responses, allowing application of other analyses not built-in into the code.

Note that while the simulator does allow specification of a very large number of scenarios, it is by no means comprehensive in its ability to simulate any type of response. Furthermore, the simulator does not allow specifying properties of individual neurons, but rather of a population. The possibilities and constraints are described below.

Note also that the simulator only includes minimal input checking and error reporting. It is the users' responsibility to enter valid (and reasonable) values into the interface.

Starting the simulator and defining the data directory

To run the temporal simulation, add the neuronal_response_simulator folder to the MATLAB path, and type neuronal_response_simulator at the command line. This will open the neuronal simulation interface (shown below). On the first time of running the simulator, the user must specify a data directory using the **Data directory** button. This path will contain all the simulated data and analysis results. Once defined, the simulator will save the path.



The *neuronal_response_simulator* GUI

Defining response properties

Each simulation involves a neuronal population. A population is defined by several general properties, responses to stimulus A, and responses to stimulus B. Responses to stimulus B are defined with respect (i.e. relative) to those for stimulus A. The number of neurons in the population is not specified directly, but is rather determined by the product of the number of different response start time, and the number of different durations.

1. The **General simulation properties** panel.

This panel includes the following data fields:

Temporal resolution: Defined the bin size for the simulation (in ms). This value must be small enough to ensure that the firing probability in each bin will be smaller than 1. The firing probability is defined by the (rate in Hz) x (bin size). The bin size also limits the maximum firing rate. For example, a value of 20 ms corresponds to a maximum firing rate of 50Hz. Although smaller bins allow defining higher rates, they will also increase the size of the simulated data (so that unnecessarily small bin sizes should be avoided).

Number of trials: The number of trials created for each neuron to each of the two stimuli.

Trial duration: Trial duration in seconds.

Stimulus onset time: Defines the earliest time at which a neuronal response can begin.

2. The **General neuronal properties** panel.

This panel specifies general properties of the neuronal responses. It includes the following data fields:

Baseline Rate range: min and max Baseline rate range determine the baseline rate of the population. If min and max define a range (that is, if max > min), then baseline rate for each neuron will be sampled uniformly from that range. If max is equal to min, then the baseline rate will be determined by this value. Note that once drawn, the baseline value remains fixed for the neuron, across all trials and both stimuli.

Trial to Trial start time jitter: Specifies the trial to trial jitter (in seconds) in start times. If the jitter is set to 0, then responses of a given neuron to a given stimulus will always begin at the same time. It is important to understand that response start time is not explicitly the time of action potential firing, but rather the time at which the underlying rate function changes. The jitter refers to trial to trial variability in the onset time, or duration of the underlying response rate function.

Specifically, if the jitter is larger than 0, the response in each trial will be sampled from an interval defined by the nominal start time (see below) and the jitter range. For example, if the start time is defined and 5, and the jitter is defined as 2, then each trial's start time will be sampled from the interval [5-7].

Trial to Trial duration jitter: Analogous to start time jitter, but applies to response durations.

3. **Stimulus A response** panel

This panel sets the properties of the neuronal responses of all the neurons in the population to stimulus A. It includes the following data fields:

Stim A response start time range: The *min* and *max* values, together with the *number of start times* field, determine the range of start times that will be applied to the neuronal population. The start times are determined by dividing the range defined by *min* and *max* to *number of start times* values. For example, if the min value is 0 and the max value is 5, and the number of start times is 6, then neurons in the population will have start times of [0 1 2 3 4 5] seconds. Note that the min and max value can be identical. For example, if they are both equal to 5, and the *number of start times* is 4, then this specifies 4 start times, all of which are 5 s.

Note that the number of start times, together with the number of durations (see below), determine the number of neurons in the population (see below).

Stim A response duration range The *min* and *max* values, together with the *number of durations* field, determine the range of durations that will be applied to the neuronal population. The definitions are analogous to those of the start time range.

Note on the total number of neurons in the population: The total number of neurons is the product of different start times and different durations. For example, if there are 5 different start times and 5 different durations, there will be a total of 25 neurons, each with a different combination of durations and start times. Note that it is possible to introduce redundancy into the population, but setting the *min* and *max* to the same values and specifying several values to span the range. For example, if *min* and *max* start times are 0 and 4 and number of start times is 5, while the *min* and *max* durations are 5, and the number of durations is 2, the population will include 10 neurons with [response start times, durations] given by: [0,5], [0,5], [1,5], [1,5], [2,5], [2,5], [3,5], [3,5], [4,5], [4,5].

Stim A response (Hz): The *min* and *max* values, determine the response of each neuron. Unlike the start time and durations, here, values are randomly sampled from the range for each neuron separately. It should be noted that as with the baseline rates, a value is sampled for each neuron once, and will apply across all trials of stimulus A. There is no "jitter" parameter for response magnitude.

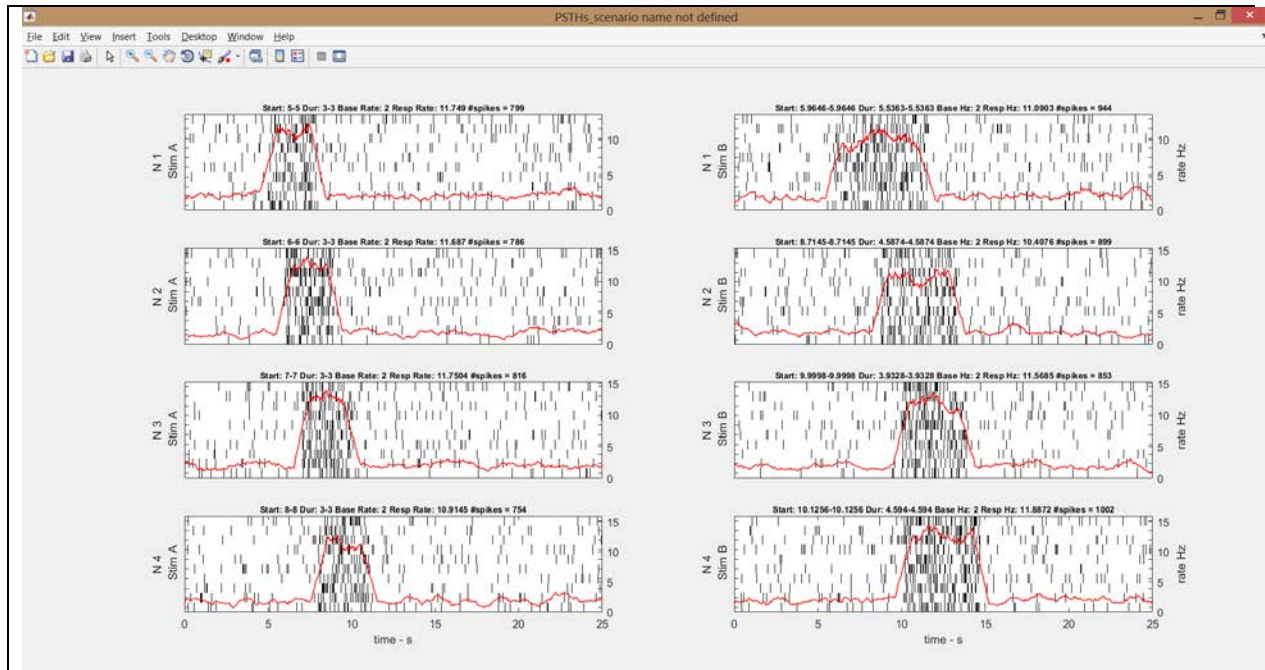
4. **Stimulus B response** panel

This panel sets the properties of the neuronal responses of all the neurons in the population to stimulus B. Stimulus B responses are defined with respect to stimulus A. They include the following data fields:

Stim B response start time range, relative to A: The *min* and *max* values determine the range of start times of each neuron to stimulus B, relative to its respective response to stimulus A. More specifically, the response start time of each neuron to stimulus B is given by its response start time to stimulus A plus a value uniformly randomly drawn from the interval [min,max]. For example, if min and max are 0, then the response start time to stimulus B will be the same as that for stimulus A. If both min and max are 3, then the response to stimulus B will be delayed by 3 seconds relative to stimulus A. If min is 0 and max is 3, then response start time to stimulus B will be delayed by some value between 0 and 3 relative to stimulus A. Note that as with other temporal parameters, the start time in response to stimulus B is fixed. Note also that the trial to trial jitter in start time applies to the responses to stimulus B just as they apply to that neuron's response to stimulus A.

Stim B response duration, relative to A: This is analogous to Stimulus B response start time, but applies to response durations. The *min* and *max* values determine the range of response duration of each neuron to stimulus B, relative to its respective response duration to stimulus A. More specifically, the response duration of each neuron to stimulus B is given by its response duration to stimulus A plus a value uniformly randomly drawn from the interval [min,max]. Note that the trial to trial jitter in durations applies to the responses to stimulus B just as they apply to that neuron's response to stimulus A.

Stim B response relative to A (Hz): These values set the response magnitude of each neuron to stimulus B, relative to its response to stimulus A. For each neuron, a value is selected uniformly from the interval defined by [min, max]. As with other definitions, it is possible to set the response to stimulus B to be identical to A (by setting min and max to 0), set it for a fixed difference from A (by setting min and max to the same value), or by defining a range of possible differences. It is the user's responsibility to avoid a situation in which the response to stimulus B yields a negative firing rate. For example, if the response to stimulus A is [5,5] and the response to stimulus B is [-10,10], this can yield neurons with negative responses. This will result in invalid values for rate (firing rate probabilities) and the simulation will generate erroneous results.



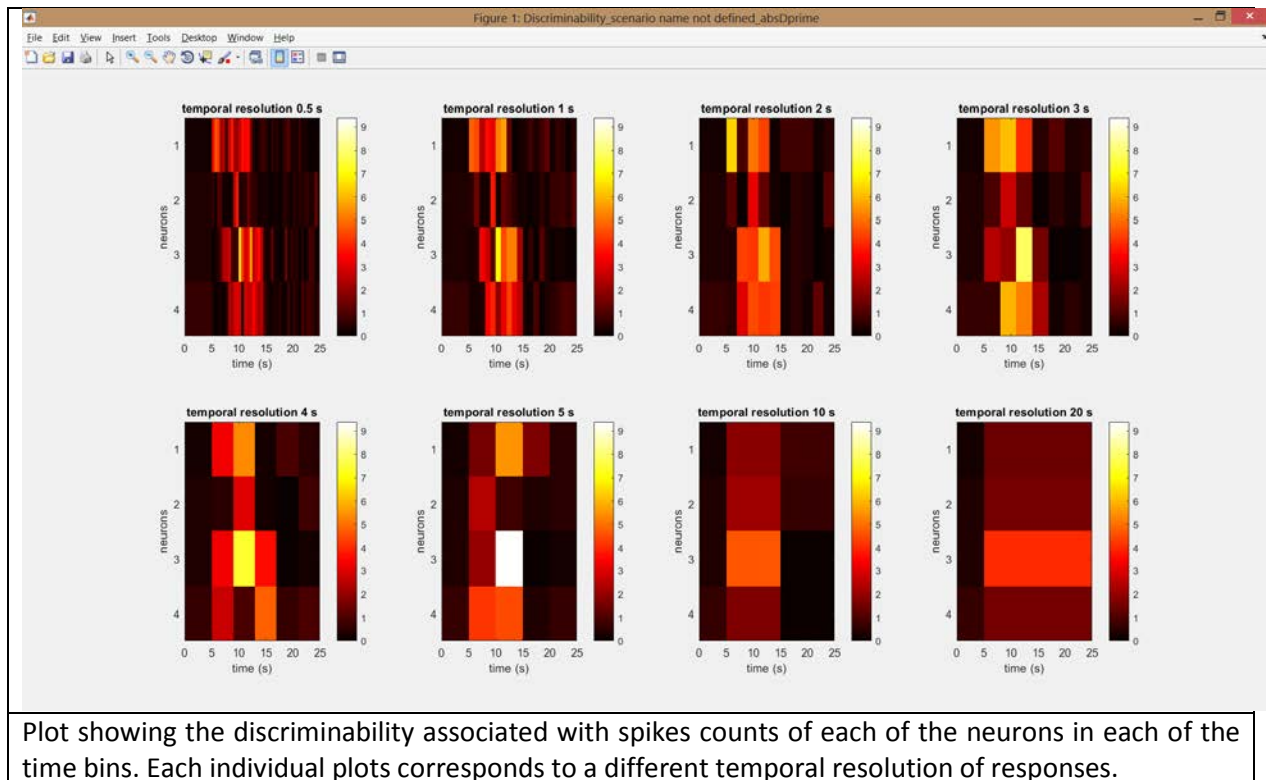
Raster displays and PSTHs for each of the different neurons for the two stimuli. The left column shows responses to stimulus A while the right column shows responses to stimulus B. The mean firing rate, (PSTH) is shown by the red trace. The title of each panel show the response parameter associated with this neuron-stimulus combination. Due to resolution limitations, the images do not always display all the spikes. To this end, the number of spikes in each data matrix is indicated in the title.

Viewing responses (neuronal activity and discriminability)

Once a scenario is defined, it is possible to view neuronal response of the population, and to view the discriminability it allows between two stimuli.

The **Show PSTH** button will create a figure raster displays and peri-stimulus time histograms (PSTHs) of each neuron in the population to each of the two stimuli. The specific parameters associated with each of the neurons will also be shown in each panel of the plot. If the number of plots is large, then to save space, titles will not be shown. The figure below shows one example of such a plot, created by pressing the Show PSTHs button. Note that the PSTH itself (red trace), is smoothed. The size of the smoothing window (in seconds) is determining by the smooth field in the View panel. **Note also that each time that the Show PSTH button is clicked a new population and a new realization of responses of this population is drawn.**

The **Discriminability** button creates plots that show the ability of the population to discriminate between the two stimuli. Five different measures, selected by the dropdown menu below the discriminability button can be plotted. These are the d' , (also known as the sensitivity index), the absolute value of d' , the auROC score, (defined as the area under the receiving operating curve – 0.5), the absolute auROC score, and the raw difference in mean spike counts. These values are color coded, and shown for each neuron, and each time bin separately. The figures show 8 different plots for different bin sizes (temporal resolutions). For example, the figure below shows that absolute d' measure applied to the PSTHs shown above. It can be seen how the discriminability changes as a function of time, for the different neurons.



Differences between deterministic and random features of the population

Taken together, values in all these panels (General stimulation properties, General neuronal properties, Stimulus A responses, Stimulus B responses) specify a *scenario*. Note that this specification is *not* deterministic, since there can be multiple realizations of any given scenario. In fact, there are two levels of randomness. The first applies to the generation of the population itself. Although start times and durations of responses to stimulus A are deterministic, response start times and durations to stimulus B may be probabilistically defined relative to A (if these value are defined by ranges). Furthermore, response magnitudes to both stimulus A and B, if defined by ranges (rather than a single value), are not deterministic. The second level of randomness stems from the fact that even when all response features are fixed, there can be different manifestations of each response. This is the case even if all jitter values are set to 0, because even for a given rate and duration, there will be a different realization of the underlying Poisson process. Clearly, if the jitter is not 0, then there will be larger differences between each realization. **These random factors imply that if the Show PSTHs or the Discriminability buttons are repeatedly pressed, the outcome may (slightly) differ each time.**

Saving a scenario

Scenarios, defined by the values in the simulation GUI, can be saved (whether they have been plotted or not) for further analysis later. When the **save scenario** button is pressed, the user is prompted to provide a name for it (for example, S1). Then, three things occur:

1. The simulation parameters, which are saved in a matlab data file named after the scenario. For example, if the name is "S1", the file will be called "S1_params.mat" and will be saved in the directory "S1", under the user specified data directory.

2. The simulation generates an instantiation of neuronal activity based on the scenario (the same neuronal activity which is displayed in the draw PSTH figure). The activity is saved in a matlab data file, which is named after the scenario, for example if the name is "S1", the file will be called "S1_spikes.mat" and will be saved in the directory "S1", under the user specified data directory. The format of this data file is described below.

3. Prepare the data for classification analysis. This involves reading the neuronal response data, (from the spike files, created in the previous stage), and creating many matrices, each with spike counts in specific windows, defined by their latency and duration and the stimulus that evoked them (A or B). These matrices are required for the classification stage, and will not be explained here in detail. See the m file `generate_simulated_classifier_struct_gui`, to learn how the data is reformatted.

While all 3 stages are automatically performed, they are not all necessarily relevant for all purposes. In fact, scenarios can be visualized without saving. However, the data created in the second stage is required to conduct custom analyses on the simulated data, while the files created in the third stage are required for classification analyses using the built in functions in the GUI.

Loading scenarios

Once defined, saved scenarios can be loaded into the interface for further analysis and modifications. To load a previously saved scenario, press the **load scenario** button. When pressed, the program will prompt for a directory, as created during the saving stage. In other words, the directory name corresponds to a previously saved scenario name. Once loaded, all the values which were associated with the scenario are loaded into the interface (the GUI simply reads the params data file which was created during the first stage of saving the scenario).

Format of the spikes.mat file

The data in the spike file (created in the second stage, after saving a scenario) can be used for any type of decoding or other analysis. The spikes file contains three variables:

1. `a_SPKS`: a 1 x n cell array, where n is the number of neurons in the population. Each of the n cells contains a p x m logical matrix, where p is the number of trials, and m is the number of time points. Each element in the matrix contains a 0 or a 1, depending on whether there is a spike (1) or there is no spike (0) in the response of the specific neuron to stimulus A in the corresponding bin. The bin is defined by the trial number and the time.
2. `b_SPKS`: analogous to `a_SPKS`, but describes responses to stimulus B.
3. `TV`: a 1 x m double array containing the time (in seconds) associated with each bin in the matrices.

Classifying stimuli

The classify panel contains controls that allow classification of the data. This analysis relies on data matrices generated during the **save scenario** stage. The details of the classification procedure are complicated and will be described here only briefly.

Classification scheme: There are two supported classification scenarios. In time sensitive classification, the data from each neuron is separated into distinct time bins (aligned on stimulus onset), each of which

is considered separately. In time invariant classification, classifiers can receive values from any time bin following stimulus onset, and must be able to classify the data without knowledge about the time from which the data was sampled.

In both schemes, classification is implemented at several temporal resolutions (by default the values are 0.5,1,2,3,4,5,10,20). The view/edit button calls the classification function in the MATLAB editor, allowing modification of these and other definitions.

Classifier type: perceptron or support vector machine classification (with a quadratic kernel).

Shuffle: provides various options for shuffling the data prior to classification. These features are relevant for the time sensitive classification.

None: no shuffle

shuffle time windows in all data: shuffle time bins (preserving neuron identity) across training and test sets

shuffle time windows in test set: shuffle time bins (maintaining neuron identity) in test set

shuffle_neurons in all data: shuffle data from different neurons (preserving temporal order) in test set and training set

shuffle neurons in test set: shuffle data from different neurons (preserving temporal order) in test set

shuffle test set: complete shuffle of test set, across neurons and time points.

Ntrain: number of trials in the training set

Ntest: number of trials in the test set

Constraints on ntrain and ntest:

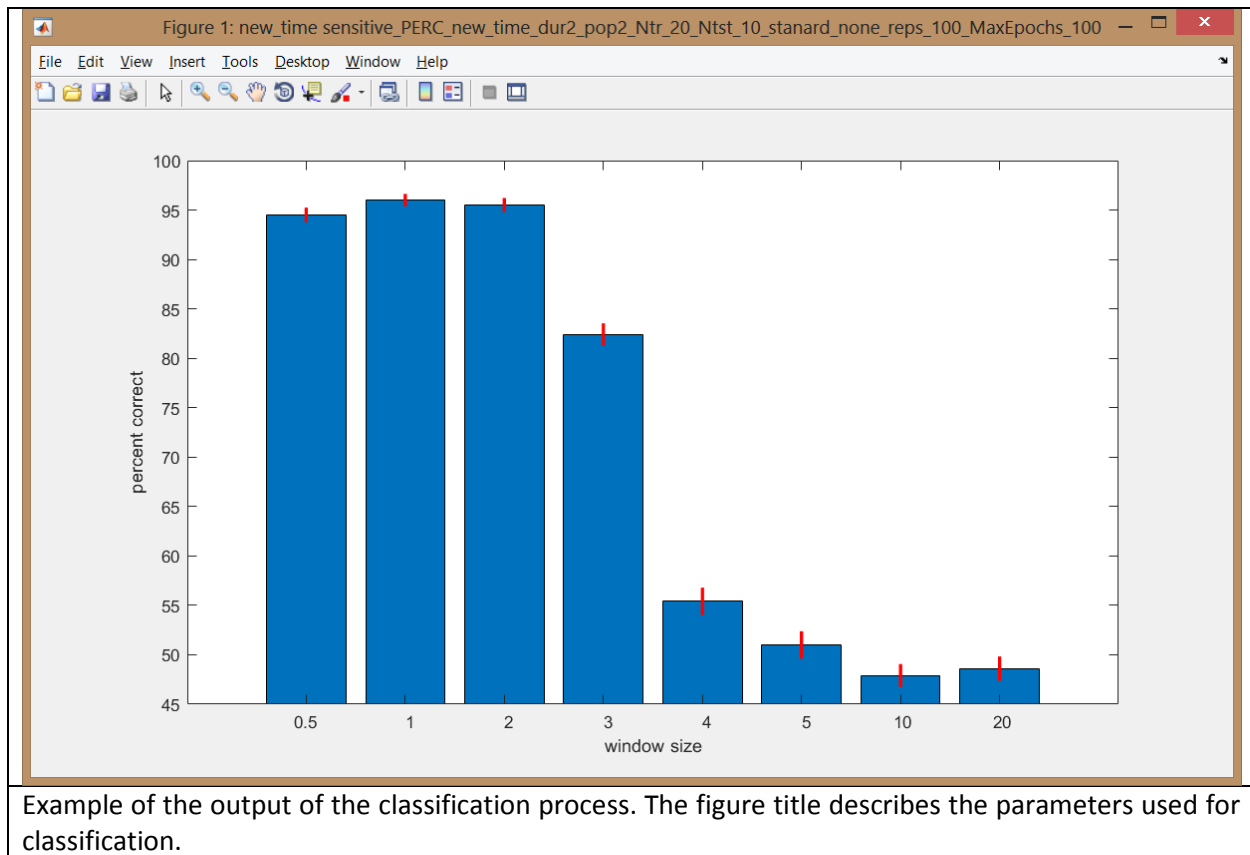
ntrain + ntest cannot be larger than the total number of simulated trials x 2.

Ntrain and ntest must be even. Half of each set will be taken from stimulus A responses, while the other half will be taken from stimulus B responses.

Max epochs: maximum number of iterations before perceptron classifier stops training. This value is only relevant if the classifier has not reached criterion performance on the training set. This is not relevant for the SVM classifier.

Repeats: Number of iterations of each data point.

The output of the classification function is a graphical representation of the classifier performance. This figure is saved in the simulation data directory. In addition, the entire classification function workspace is saved in a mat file (under the scenario directory). The data file name reflects many of the key classification parameters. Press the view/edit function button to access the file for a description of the variables.



Tips on modifying the neuronal response simulator

User can view the code and modify the interface and code for their own needs. This can be best done using the MATLAB GUIDE tool to open the neuronal_response_simulator function. Key functions called by the GUI include:

sim_params = get_stimulation_params(handles): collects simulation parameters from the GUI and returns a structure with these parameters.

population = draw_population_gui(sim_params): draws a population based on the GUI specified parameters.

[a_SPKS, b_SPKS, TV] = get_response_realization(population,BS,TD,NT): draws a specific realization of a response based on the population drawn.

plot_scenario_psths_gui(population,a_SPKS,b_SPKS,TV,NT,BS,scenario,SWW): shows the populations PSTHs and raster

plot_scenario_discriminability_gui(TV,a_SPKS,b_SPKS,scenario,Dmeasure,CLims): shows the discriminability associated with each neuron and time bin.