**Javascript plots**

See the waterfall plot functions as a concrete example

Several functions should be implemented for new javascript plots: One function to retrieve data i.e. via the “get\_data” view and interface with the paneling functionality and the other to actually draw the plot. We utilize this functionality along with the summary modal tables specified by the administrator in the node\_content dict. For example a plot pertaining to gene results can be specified for a given variable by providing the appropriate javascript function in the ‘plot\_vars’ element of the node\_content->Gene->args dictionary.

The only restrictions for the data retrieval function is that it should call “network.js:add\_to\_image” and supply it with several arguments:

1. The data to be used by the drawing functionality
   1. Including a “title” key-value pair.
2. The type of plot which will be used by the paneling functions to determine which function to call to draw the plot.
   1. Note that this string needs to be specified as the key of the jsplot\_funcs object (in “custom\_plots.js”) referencing the plotting function.
3. Values which will be supplied to the history tracker. Currently not used.

The “get\_data” view is fairly flexible and allows the user to specify the handler function in the config file as part of the plot\_data\_types dict. The AJAX request needs to at least specify a ‘type’ corresponding to an entry of plot\_data\_types. The handler should take a single argument consisting of a dictionary derived from the request and should return a dictionary with at least a ‘data’ element as well as a title string.

The drawing function should expect the following:

1. A d3.js selection referencing the panel the image should be drawn at.
2. The data object
3. Width of the panel
4. Height of the panel
5. A Boolean indicating whether the shiftKey was pressed.

**External plots**