Updating NeuroShiny

NeuroShiny app requires several files to run. Outline below is how to create each file if it needs updating. Basic R knowledge is required to do this.

You will need to change the setwd() code to your directory with the NeuroShiny files.

**Split Data by Gene**

Due to large size of many of the datasets used to generate the plots they have been split up into individual files for each gene. The datasets this was done for are;

* Benito – organoid time-course plot
* BrainSpan – region and time-course plots
* Comparison – time-course plot with all datasets combined
* Fan – scRNA-seq plot, tsne and violin plots
* Kanton – organoid time-course plot and evolution plot
* NPC – 2D differentiation time-course plots
* Pollen – organoid time-course plots (Pollen, sloan and camp) and pollen evolution

To generate these files use R code file “split-data-by-gene.R”

Raw data files to run this code are in the folder “Data Files”

If you want to edit any of the file formats generated by this code then you’ll need to edit the function make\_gene\_file\_all()

**Making Gene Info Table and Gene Lists**

A large gene info table has been generated containing data from several datasets (Nowakowski, Fan, Zhong, BrainSpan, Bhaduri, SFARI). The code used for generating this is called “make\_data\_files.R”.

This Gene Table and gene lists are required for the Explore Gene Tab – Gene Table and gene set overlap function.

Within this file multiple datafiles are collapsed and combined so that the gene table has just one entry for each gene and for each dataset information is collapsed into one cell.

If you only need to update a particular section you can run that bit of code and skip the other sections and just load them in the section “Make Gene Table”.

*Updating SFARI Gene List*

Replace the “SFARI-Genes.csv” file with latest release (current release used 09-02-2021) and re-run make\_data\_files.R code.

*Adding new data to Gene Table and Gene Lists*

If you want to add to this data you will have to;

1. Reformat data so there is just one row for each gene – collapse data using paste() function, see what was done for other datasets
2. Update the gene\_info\_column\_lists list to include the new column names
3. Add to gene\_list if you want this new data to be used in the Gene Set Overlap function

**Cell Type Data Files and Colour Scheme**

For generating a consistent colour scheme and for making the combined scRNA-seq plot the R code file make\_celltype\_datafiles.R was used.

As each study classifies cell types with a variety of different names with have re-classified cells with a more general cell type name to generate a consistent colour scheme between studies.

*Change the cell type colour scheme*

Update the variable for the cell type for new colour. For a list of colours you can use <http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf>

View the updated colour scheme and take a screenshot of the new ledged and rename it “CellTypeLedgend.png”

*Adding new scRNA-seq study to Combined Cell Type plot*

To add a new scRNA-seq dataset to the combined plot the following needs to be done;

1. Re-classify cell types to fit with our scheme – For each cluster rename it with a Cell Type used in “Make Colour Scheme” section i.e.

new\_data\_cell\_types <- unique(new\_data$Cluster)

new\_data\_cell\_types$Cell\_Type = "Other"

new\_data\_cell\_types$Cell\_Type[new\_data\_cell\_types$Cluster %in% c("ExN","ExDp1"] = "Excitatory Neuron"

colnames(new\_data\_cell\_types) <- c("Cell Subtype","Cell Type")

See what was done for other datasets.

Merge new cell type reclassification with other datasets

cell\_subtype\_df <- dplyr::bind\_rows(nowakowski\_cell\_subtype\_df, fan\_cluster\_group,

polioudakis\_cell\_types,zhong\_cell\_types, new\_data\_cell\_types)

1. Reformat cluster enrichment data file

Reformat data files so it is in the format (cluster is original cluster name);

colnames(new\_cell\_plot\_data) <- c("Gene","Pval","Fold.Change","Cluster")

1. Merge and save new files

cell\_plot\_data <- dplyr::bind\_rows(cell\_plot\_data1, cell\_plot\_data2, cell\_plot\_data3,cell\_plot\_data4, new\_cell\_plot\_data)

Once all these new additions to code are added run whole code to make new file.

1. Update the Table below plot

To add new dataset to table open excel file Datasets and add new info to the table.

To add symbol in shiny app.R file go to;

output$DatasetSummaryTable <- renderTable({

**Key <- c(as.character(icon("circle")), as.character("&#9650"), as.character(icon("square")), as.character(icon("plus")))**

dataset\_table <- cbind(Key, dataset\_table)

dataset\_col\_names <- colnames(dataset\_table)

dataset\_table <- as.data.frame(lapply(dataset\_table, as.character)) #Convert to character to round numbers

colnames(dataset\_table) <- dataset\_col\_names

xtable::xtable(dataset\_table)

},sanitize.text.function = function(x)x)

Add new symbol to the Key string shown in bold