# Input format requirements:

factors.tbl must be a tab-separated file, organized as follows (example included in data file):

* first column:
  + firstline is header, ”file”
  + each row is the name of each file as named by the RSEM algorithm (default naming is important to determine isoform vs gene level summary for this analysis)
* second column:
  + firstline is header, “replicate”
  + each row is a string representing which biological replicate corresponds to the data in each file in the first column. These must be matched between the input and IP/antibody samples
* third column:
  + firstline is header, “abTreatment”
  + each row indicates whether or not the sample for the row was treated with antibody pulldown (enrichment for 8OG) corresponding to the file in the row
  + It is REQUIRED that each control be indicated with an entry of “none”, but the indicator for antibody can be any string (i.e. “yes” or “8OG” or “m5C” in case multiple antibody pulldowns were performed, for example)
* fourth column:
  + firstline is header, “exposureCondition”
  + each row indicates the factors that were used for exposure study corresponding to the file in the row
  + It is REQUIRED that each control be indicated with an entry of “control”, but the indicator for other treatments can be any string (i.e. “exposed” or “test” or “ozone-acrolein-methacrolein” or “OAM”)
* fifth column:
  + firstline is header, “exposureLevel”
  + each row indicates the level of exposure corresponding to the file in the row (i.e. “Low” or “High” or “400ppb” or “0.4” or “4”
  + If you only have one exposure level, you must still include this column and just enter all the same value for every row. It can be arbitrary, as it will not be used in DESeq analysis and only in organizing data
  + The variables in this column will be used to organize your data into subsets for each exposure level for enrichment analysis

# Notes to self on dev plans/needs:

* add functionality for other count tables besides RSEM
* rework it all to be in line with my requirements for factors.tbl
* add processing of path so dont have to change wd partway through
* finish setddsnames in line with new requirements
* build initial formatting/error checking into all functions
* write the script for running all these functions, vignette?
* write all documentation Roxygen stuff
* for splitandbuilddeobjs function: need to add a test and warning for if a factor only has one member of a level, and drop it, i.e. the R5 sample
* is “invisible” working for the txiload function? Also, double check that the folder/file naming works correctly, I think at this point it does but doublecheck
* loaders.R file:
  + #note to self, I think I'd like to make this figure things out on its own the ordering, rather than feeding numbered rows into the DEseqdataset constructors, so to that end I'll have the factors.tbl first column be the filename, then the remainder be the factors themselves.
  + #will probably need rigid column naming for the factors though...
  + #note to self, can't find factors.tbl that is read in my real dissertation work. i have output logs, result files, etc, but can't find the script and inputs for some reason...
  + In checkrownames function, #should I have this loop over all files rather than just the first, checking for consistency among files as well as the various components of the txi object?
* for the analyzers.R file:
  + #is it worth writing a dedicated function for looping over dds/ddr obj list named object entries? then that could be called by all subsequent analysis functions…
  + #make sure to add error-check for length of named list parameters?
* Finish writing all functions in vizTools.R, right now is just road map for where I’ll go with that.

Updates on 6jan2023

So I’ve decided that rshiny app will rerun all analyses, so that users can upload their own data if they want, or select the existing data (gzip compressed count tables are fine for tximport and fit under 100mb limit for rshiny app), eventually will have an option to fetch juan’s other paper data too but that’s not a now problem, it’s a later problem.

So anyway, bottom line, gonna have to put in some work on this package to make viz tools available to the shiny app, but for now have been moving forward with shiny app just getting together what I have so far in the pipeline

Updates on 29dec2022

Making Rshiny app for the data… need to run DESeq and store dataset

gotta change out enst vs hgnc, have a group lookup for all hgnc transcripts to show them all in the chart in shiny?

ok so make attributes they can fetch a searchable list from the description of the attribute, that then builds their vector of attributes from the corresponding attribute tag – then they can get BM and set up plot dynamically based on what they want to visualize?

Working on making Rshiny app, I realized that the DESeq objects are too large for inclusion in the app (current limit 100MB), so need to do one of the following:

* Have app re-run all the DESeq analysis
* I can extract data into simpler data structures abandoning the DESeq classes to shrink the object size
* I can maintain the DESeq data structure and drop some set of rows out of it – I think this is probably the best bet, drop the padj=NA rows and then the ddrObjList is small enough… trying to see differences though between the padj=NA vs basemean=0 filters…

So I’ll run through it all again, filter the ddrobjlists to have padj=NA dropped, and then from there I can select the Tfm one too… honestly, does this need to just outsource persistent data though, if someone wants to explore different ways of filtering and seeing venn diagrams accordingly… guess I need to really determine what I feel is the goal of the shiny app…

For the PCA plots, can just save the plot as an object which could be plotted with a checkbox, or something – then can skip uploading the ddsTfmList object that will also be large…

Makevenn, can have them choose a cutoff for filtering from the padj=NA filtered data already, it takes in ddr lists – in fact, let’s go ahead and have it be the case that they build the ddrSigList as part of the app!

Major concern

OK so I think Juan has a slightly different option set in the independent filtering stage, maybe, since:

for the 8OG\_OAM\_Low data with FDFT1 transcript (ENST0000052946), my analysis matches the log2FoldChange (=11.746), but mine has an adjusted pvalue of 1 and suppl data 3 has adjusted pvalue of 0.0808. See also the comment notes after the makeVenn function in the vizTools.R file for the comments on counts