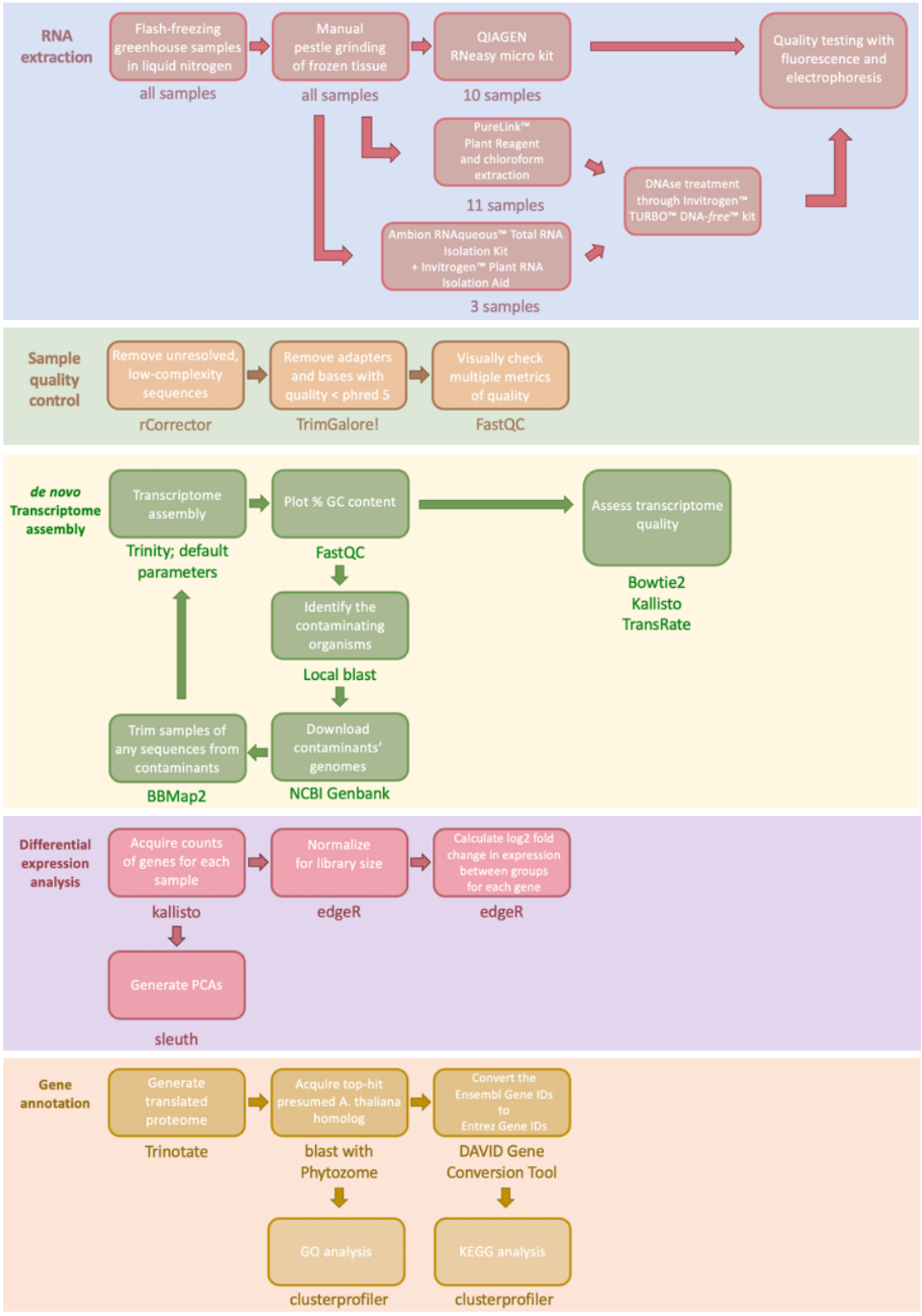
Abagail Burrus
Kramer Lab, July 7, 2022

De novo transcriptome gene annotation guide

This guide covers the orange background/tan-boxed portion of this RNA Seq workflow:



This is specifically for work with genes lacking a reference transcriptome.

If you're working in *Aquilegia*, or another comparable group, please save yourself some massive headaches and consult your Kramer lab colleagues on using agriGO here: http://systemsbiology.cau.edu.cn/agriGOv2/

(You can also probably ask them for their scripts if they had to code anything for it.)

In my case, agriGO was a broken set of pages for months with no explanation or context, so I couldn't use it.

Instead, Yan Gong in the Kramer lab helped me a lot with getting functional annotation performed through a mix of Phytozome and GO Ontology/Panther DB. This utilized *Arabidopsis* as a reference genome.

The first step in any approach is to convert your transcriptome to a proteome.

Generating your Proteome

First, I used TransDecoder to identify Open Reading Frames (ORFs) in my transcriptome, creating and running these scripts as jobs from within the directory containing my transcriptome:

transdecoder.sh:
—

```
#!/bin/bash

#SBATCH -J transdecoder
#SBATCH -n 15
#SBATCH -t 0-13:00
#SBATCH -p shared
#SBATCH --mem=37000
#SBATCH -o transdecoder.out
#SBATCH -e transdecoder.err

module purge

module load TransDecoder/5.3.0-fasrc01

TransDecoder.LongOrfs -t Trinity.fasta
```

(you sbatch the transdecoder.sh, very direct)

Then:

transpredict.sh:

```
#!/bin/bash

#SBATCH -J transdecoder
#SBATCH -n 25
#SBATCH -t 0-13:00
#SBATCH -p shared
#SBATCH --mem=37000
#SBATCH -o transdecoder.out
#SBATCH -e transdecoder.err

module purge

module load TransDecoder/5.3.0-fasrc01

TransDecoder.Predict -t Trinity.fasta
```

For my example above, my results were in
/n/holyscratch01/davis_lab/aburrus/trinity_out_dir/Trinity.fasta.transdecoder_dir
The main output, your proteome, will be "___.fa.transdecoder.pep" within that new directory (Trinity.fasta.transdecoder_dir)

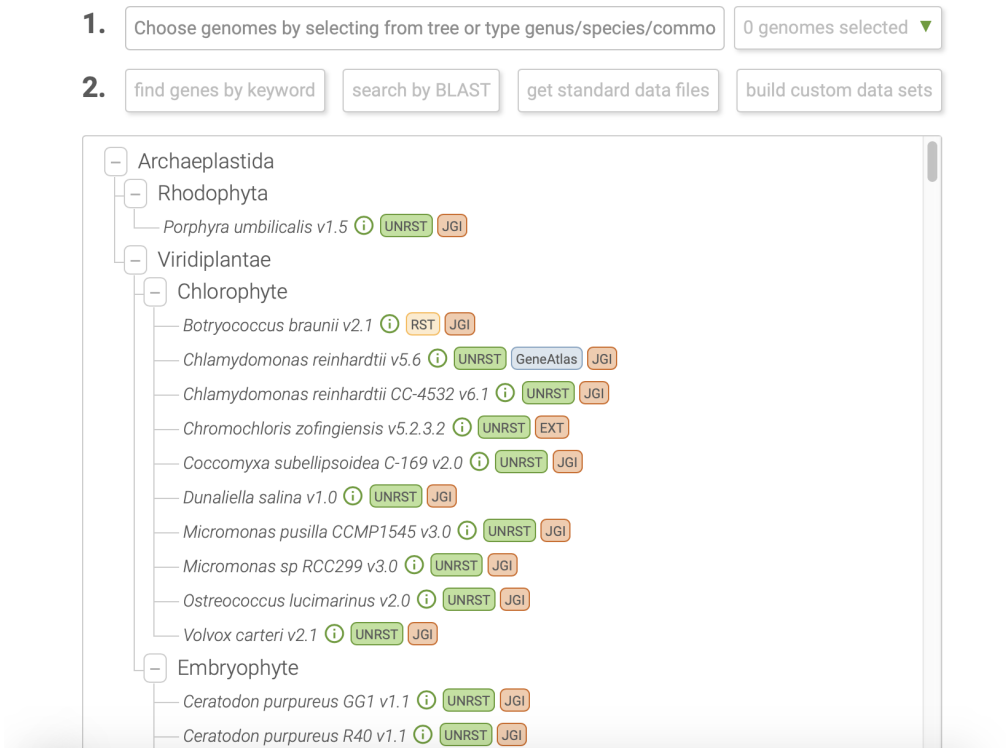
Next, we need to get the proteome's protein sequences of our DE genes and convert them into the best-identified hits of *Arabidopsis* proteins.

Generating your *Arabidopsis* putative homologs

This is another tedious step, but it's not too miserable if you don't have very many DE genes to sort through. (My biggest list was 1000 DE genes; if you're working with more than that, you should find a way to automate this process.)

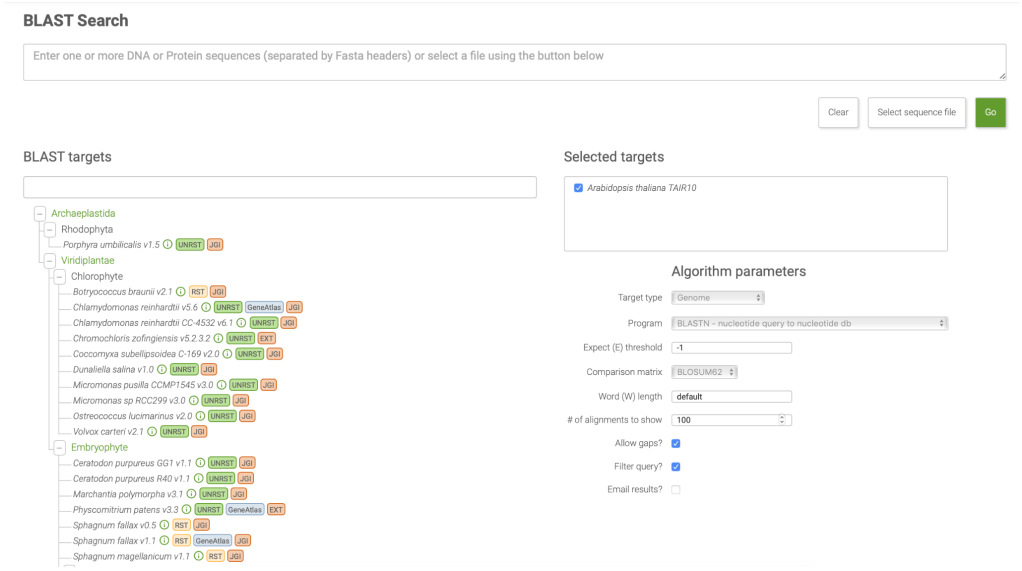
- I opened my list of significantly differentially expressed genes (like those upregulated in glandular tissue)
- Then I searched one by one for those Trinity genes (like TRINITY_DN1808_c0_g3_i1, for example) in the proteome (they'd show up as TRINITY_DN1808_c0_g3_i1.p1, for example)
- Then I pasted the found protein sequence (with its >TRINITY label too - don't forget to keep both lines of the two-line FASTA entry together!) into a new file, naming it something like "upregulatedproteinsforglands"
- Repeated with more and more of those Trinity genes, building a file listing the proteins for all the differentially expressed genes I want to annotate

I then submitted this protein list (names and amino acid chains) to Phytozome's *Arabidopsis* BLAST tool here:
https://phytozome-next.jgi.doe.gov

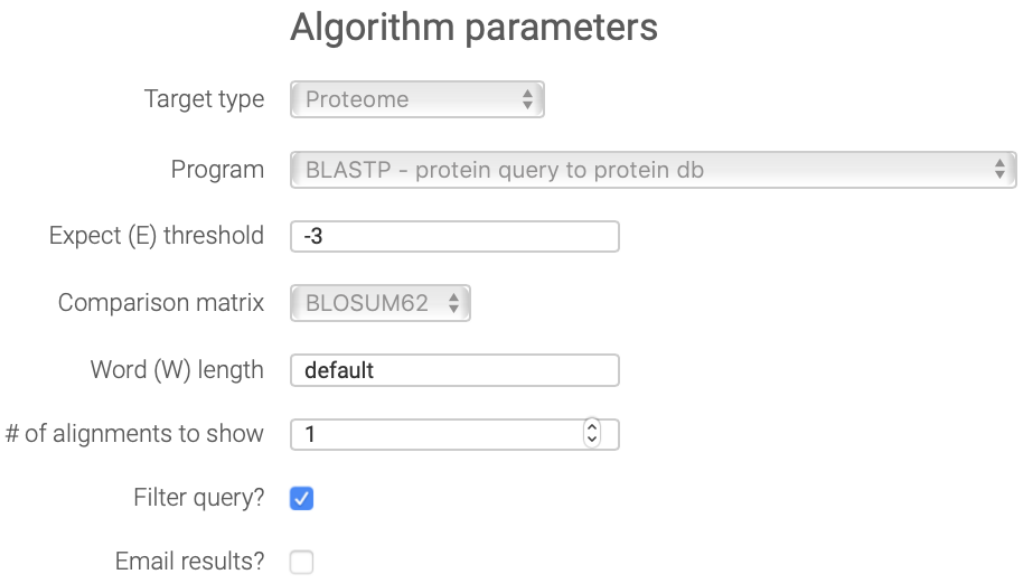


Scroll in that box to whatever reference you want (I used *Arabidopsis thaliana TAIR10*) and it should show "1 genome selected" next to box 1. above.
Then for the box 2. selection, click "search by BLAST"

Now you see a new page:



What you'll mess with now are the algorithm parameters you see on the right. If your browser for whatever reason de-selected your reference genome, click it again in that big phylogeny (again, I went with *A. thaliana*), and consider the hits you want from this protein search (selecting "Proteome" for target type and "BLASTP" for Program):



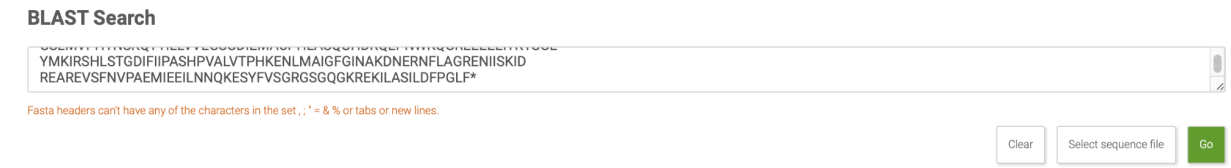
This set-up I've customized above will put a threshold of significant protein similarity to 1E-03 (the E threshold), and it'll show only the top alignment hit. This is, from what I can see in the literature, fairly standard.
You're welcome to get more hits returned (# of alignments to show), but keep in mind that this may greatly increase your resultant file size as well/volume of protein data you have to annotate and sort through.

The emailed results are convenient, but I also just exported my results in a table, so it's fine either way.

Now that you have customized the algorithm, you need to submit your protein sequences that you translated for your DE genes.
Phytozome is picky in this regard!
If you directly paste your results from your protein text file (an example here below,) you'll get errors.

```
>TRINITY_DN51849_c1_g1_i1.p1 TRINITY_DN51849_c1_g1~~TRINITY_DN51849_c1_g1_i1.p1  ORF type:internal, len:136 (+)score=14.27
TRINITY_DN51849_c1_g1_i1:1-405(+)
AMGLGADGVSIGTRFVATEESYAHPRYKQLIVEHDKTEYQNVYGRSTWPNSAQRILAIGA
GPFFKRLPPPETDEGQPSIGTATVWGEHKEITYYSEIPPNETATGDFKLMALYCGEGVGL
ITKILTVQEVMDQIV
>TRINITY_DN8478_c0_g1_i1*****
>TRINITY_DN1006_c0_g1_i1.p1 TRINITY_DN1006_c0_g1~~TRINITY_DN1006_c0_g1_i1.p1  ORF type:complete, len:277 (-)score=44.32
TRINITY_DN1006_c0_g1_i1:2603-3433(-)
MSLRLITQVINHGVPERLIESMIEFCGEFFDLPEEDKQEYQGKHVLDPVRFGTSFNASVE
TVHFWRDYLKVFVHPEFHSPSKPSGFRECSLEYSRRVREVAREVLKGISESLGLEPNYID
KAMKLDNGLQILAANYYPPCPQPELAMGMPPHSDHGLLTFLVQNGSNGLQIHHDGKWINI
QATPNAFLVNTADHLEILSNGNYKSVLHRAVVNSKATRISIAMATGPSLDSVVSPAQELL
DGEQTSAYMGIQYKDYLEFQQSNKLDGKSGLYRLRT*
>TRINITY_DN1995_c0_g1_i25.p1 TRINITY_DN1995_c0_g1~~TRINITY_DN1995_c0_g1_i25.p1  ORF type:5prime_partial, len:531
(+)score=98.67 TRINITY_DN1995_c0_g1_i25:3-1595(+)
TISLSLSSSFETKPKSSNTNMFIENFKVESPNVKYTENEIHSVYNYETTELTHENRNGTY
QWVVKPKTVEYQFKTDIRVPKLGVMLVGWGGNNGSTLTAGVIANKEGISWATKDKVQQAN
YFGSLTQASSIRVGSFNGEEIYAPFKSLLPMVNPDDIVFGGWDISDMNLADAMARAKVLD
IDLQKQLRPYMESMVPLPGIYDPDFIAANQGSRANNIVKGTKKEQVDQIIKDIREFKEKS
KVDKVVVLWTANTERYSNVVVGLNDTMENLLASVDKNEAEISPSTLYALACVLENIPFIN
GSPQNTFVPGLIDLAIQRNCLIGGDDFKSGQTKMKSVLVDFLVGAGIKPTSIVSYNHLGN
NDGMNLSAPQTFRSKEISKSNVVDDMVSSNGILYEPGEHPDHVVVIKYVPYVGDSKRAMD
EYTSEIFMGGTNTIVLHNTCEDSLLAAPIILDLVLLAELSTRIQLKAEGEGKFHSFHPVA
TILSYLTKAPLVPPGTPVVNALSKQRAMLENILRACVGLAPENNMILEYK*
>TRINITY_DN24634_c1_g1_i1.p1 TRINITY_DN24634_c1_g1~~TRINITY_DN24634_c1_g1_i1.p1  ORF type:complete, len:145 (+)score=25.46
TRINITY_DN24634_c1_g1_i1:27-437(+)
MGPDIGGPKLVAAAVNAGGSGFLRVPETNSEATAEKLRQAIKLTKQRCTGKFGVEIVLKF
AYSYHWEVIKKERVAILGTYWGYPSKELVDEAHANGMKVVSQIGTLKEAQNCIASGFDGL
IVQCYEAGGHVRGQEA*
>TRINITY_DN6924_c0_g1_i6*****
>TRINITY_DN28888_c0_g1_i3.p1 TRINITY_DN28888_c0_g1~~TRINITY_DN28888_c0_g1_i3.p1  ORF type:complete, len:593 (+)score=74.95
TRINITY_DN28888_c0_g1_i3:265-2043(+)
MSTIIIFLALTSLLCITAYKIPYPESDLEVCVRSCQRTEQDLSGLVTCTSRCKQIHESSR
SPRWEIERCQERCRIEERGRGEVQRCEQRCEEEVIKRRKQEQQRGEEKQHHEHERERERG
GNEEDPQWMYKKCLEICSKQSPRQEWSCKRRCTRQLEEEGGKGKRHSLTESREQRDNPYH
FRYDERMKTWFRSQESQLKVLDKFSDKSDLLRAIENYRLAVFEANPNTFLVPHHSDTEDI
YVVLNGKGILSYVSGEKRVSHNLERGDVIWIPAGAHRYWINHDSRQKLRVAILINLVNGA
LGNLKAYFSVGCGKPNSFYSVFSDNILEAAFNTPKHQLDRFIGQQKQGPVLKVPSEKLEA
LTKHSTSTKGRGRSAWDPFNLLQEQPVYSNNHGKFHEALPKDREQLQNMDVSVSFTNITQ
GSLMVPHYNSKQTTILLVVEGSGDIEMACPHLASQGHDRQEPNWKQGREEEEEITKTSGE
YMKIRSHLSTGDIFIIPASHPVALVTPHKENLMAIGFGINAKDNERNFLAGRENIISKID
REAREVSFNVPAEMIEEILNNQKESYFVSGRGSGQGKREKILASILDFPGLF*
```

^Example file



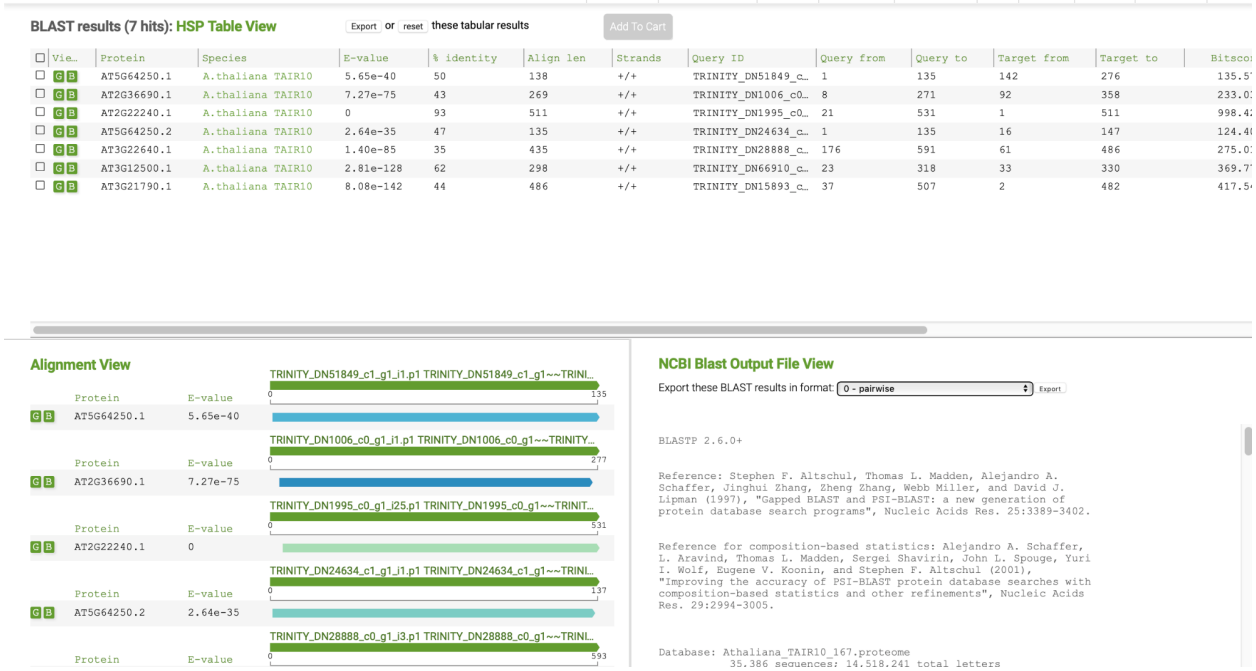
^Error you get if you paste those results in

To fix this, you need to remove the commas (,) and equal signs (=) that the protein sequences have in their headers (I don't remember how I figured out these were the specific problems).

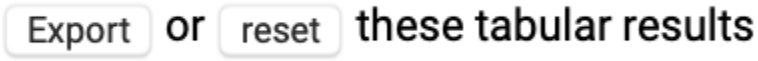
I use "find and replace" search and replace , with nothing (putting nothing at all in the "replace" part), then do the same with = (example below)

Pasting these edited sequences into the BLAST search box shouldn't have the error now. And then you click "Go"!

The results look like this:

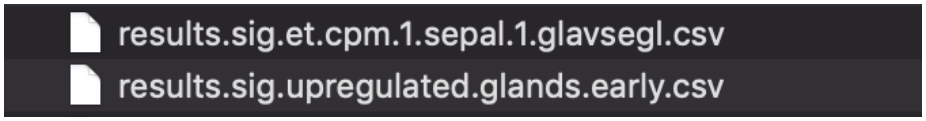


And you can export the table results along the top of the page:



Now you just need to add functional annotation to this list of proteins.

I recommend you bring up your significant results table of DE genes from the DE analysis (examples of what my files were named; if you used the edgeR script in the DE guide, yours probably looks similar:)



Now this is *totally optional*, but I think it would be a good idea to append your protein labels (AT5G64250.2, AT3G12500.1, *etc.*) to your DE genes listed. This way, you can directly reference the log2 fold change of your DE genes with the annotated protein information attached. You could also repeat this later with functional category information (GO output), if you want..

If adding in the protein labels is tricky and you're not sure how to navigate the phytozome output's order of results being different from your DE list's results, here's how I did it:
From the exported protein list, you need to copy two columns:
"Protein" and "Query ID"

Paste these two columns into your results.sig.et. list of results, but off to the right from the rest of your results' columns.

Then insert a new, empty column to the right of the k.gene.list (gene identifiers, example shown below) or to the right of the logFC (also shown below):

results.sig.et.cpm.1.sepal.1.glavsegl

k.gene.list	logFC	logCPM	PValue	FDR
TRINITY_DN23920_c0_g1_i17	-6.9199268194774000	5.733538984257560	3.21352489067044E-37	1.93441344318798E-32
TRINITY_DN2151_c0_g1_i2	-13.36337693252660	4.583164176311890	7.41477594321553E-35	2.23169926338901E-30

I used a special formula to insert the protein name according to its corresponding Query ID, which should be identical to your k.gene.list
(If your Query ID is really long and your k.gene.list is short, you can split your Query ID into columns through the Data tab of Excel, grabbing only the first chain of characters before the space- that first chain should match the k.gene.list information)
=VLOOKUP($B2,$Y:$Z,1,FALSE)
=VLOOKUP (lookup_value, table_array, column_index_num, [range_lookup])

So, in this case:
$B2 is the column (and cell) for the gene name we want to find the protein for (like TRINITY_DN23920_c0_g1_i17 above),
$Y:$Z is the range of the protein table we pasted in (the Protein and Query ID columns),
1 is the second column in that range (customize this to whatever column in your range above has the proteins you want - in my case, it was 1, but if my pasted protein information had been Query ID and THEN Protein, I would have 2 in this place)

The k.gene.list in the picture above is my Column B,
This is an example my pasted protein info from the Phytozome results, into Columns Y and Z:

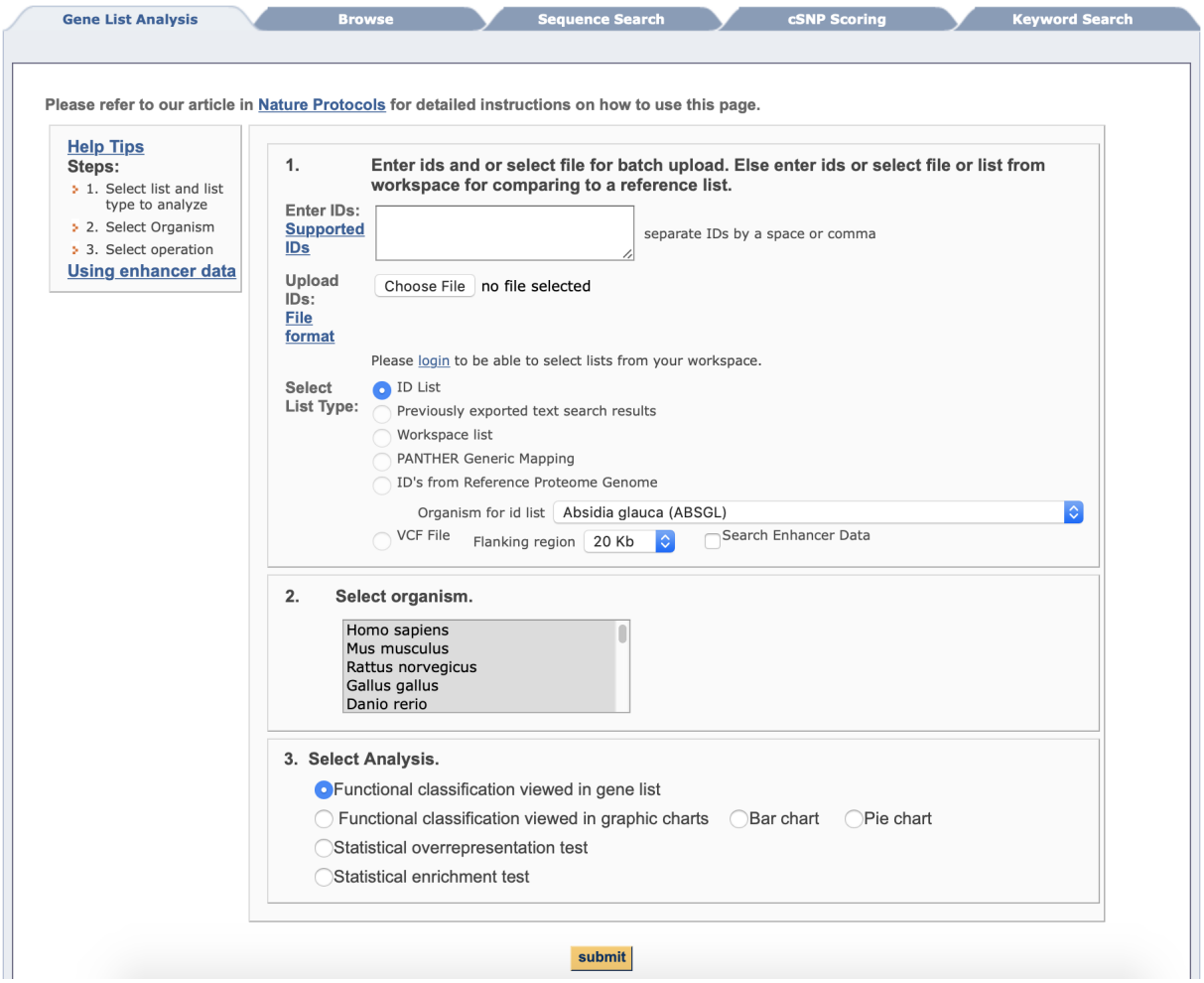
Y	Z
Protein	Query ID
AT3G03380	TRINITY_DN1825_c0_g1_i2.p1
AT3G22380	TRINITY_DN1825_c0_g1_i2.p1
AT4G31940	TRINITY_DN1825_c0_g1_i2.p1
AT1G11260	TRINITY_DN12363_c0_g1_i17.p
AT5G14940	TRINITY_DN10590_c0_g1_i21.p
AT5G49360	TRINITY_DN5625_c1_g1_i62.p1
AT1G29940	TRINITY_DN1236_c1_g1_i19.p1
AT4G10120	TRINITY_DN2114_c0_g1_i19.p1
AT2G38280	TRINITY_DN7168_c0_g1_i5.p1
AT4G27290	TRINITY_DN9234_c0_g1_i11.p1
AT3G47990	TRINITY_DN11294_c0_g1_i12.p
AT4G37800	TRINITY_DN4459_c0_g1_i21.p1
AT4G34530	TRINITY_DN15667_c0_g1_i2.p1

And Column C is the new, empty inserted column where I input my formula.
(One you've done this for your first results row, C2, you can drag the formula down to autofill the rest of your results)

Acquiring annotations

The annotations are in this case the putative homologs in *Arabidopsis* that correspond to the *Arabidopsis* proteins we got in our proteome/Phytozome work.
(Yan Gong helped me a lot with this part too!)

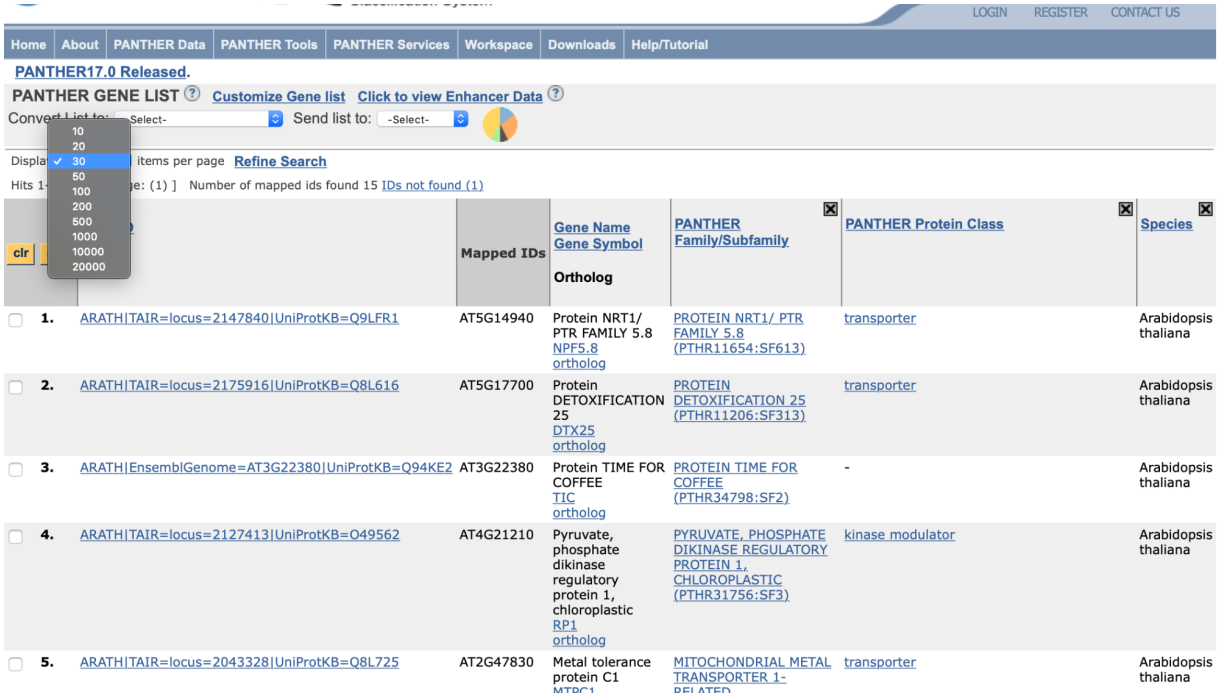
Navigate to: http://pantherdb.org
You should see this input menu already open:



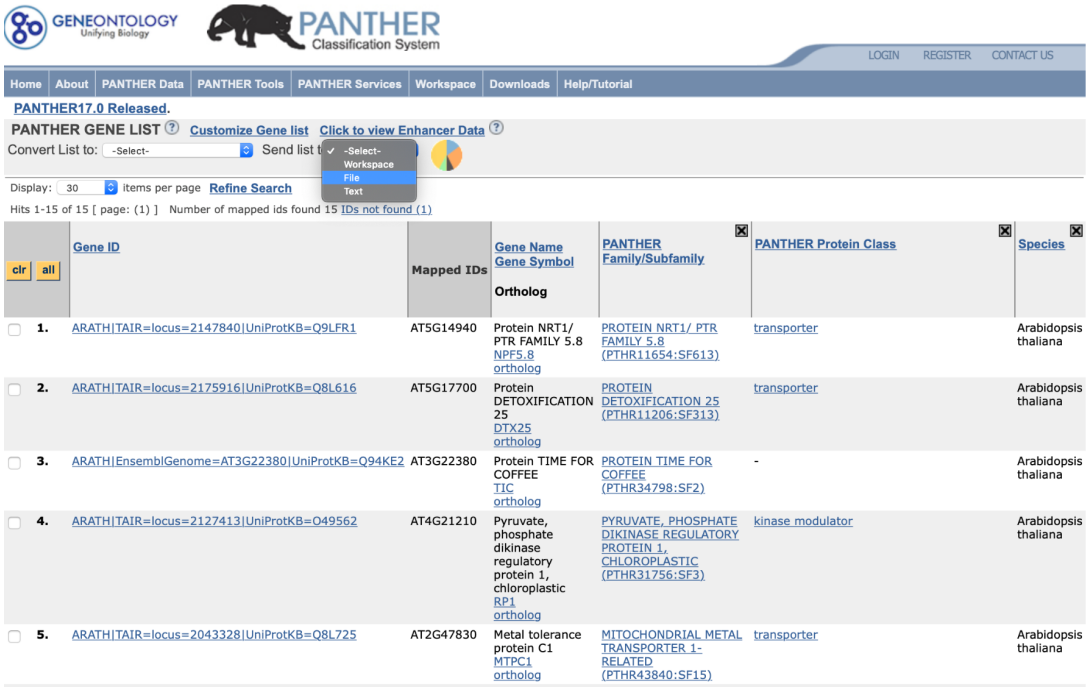
In (1.), paste the Protein list from your DE+Phytozome results (the AT5G64250.2, AT3G12500.1, *etc.*)
In (2.), select whatever your reference for the proteome conversion was (in my case, I scroll to click *Arabidopsis thaliana*).
In (3.), I just went with the first option - we'll work on overrepresentation (enriched pathways) later, but for now we want the common names of the homologs.

Your results should be a table with a bunch of gene names produced - it might be a very big list!

To export this list as one cohesive table, you need to first maximize the results shown at once:



In this drop-down menu, I'd pull it down to 20000 or whatever's higher than your results list. Now when you export the file to text, it'll include all results available (rather than just being a table of the first 20 results or something like that).
The "Gene Name / Gene Symbol / Ortholog" column is what you're most interested in here.



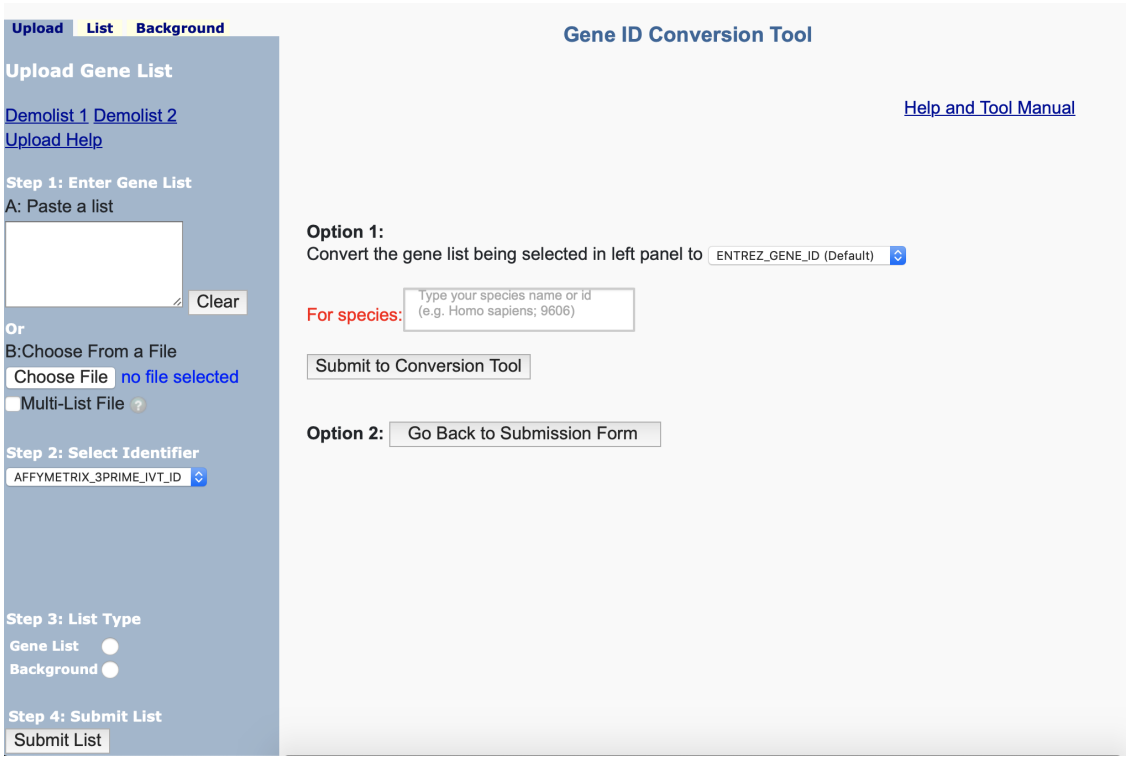
^This "sending to file" is how you get your results.

Finding overrepresented/enriched functional pathways for your DE genes

It's very interesting to see what sort of functions your differentially expressed genes might be fulfilling. If we look at genes differentially expressed between a leaf and a stamen, for example, I'd guess that we would see for our leaf an upregulation of genes with believed functions in photosynthesis. If we found specific DE genes known to control photosynthetic pathways in our Panther work above, that would be very cool, but these pathways are another source of information.

To easily assess the gene ontology (GO) at work in our DE genes, the first major step for me was getting Entrez ID names corresponding to my *Arabidopsis* protein names. The package we use for the GO work has some functions that rely on this alternative naming framework, for whatever reason.

I converted my *Arabidopsis* protein names (which are ensembl IDs) to entrez IDs through the DAVID tool:
just convert your ensembl IDs to entrez IDs through the DAVID Gene ID Conversion Tool:
https://david.ncifcrf.gov/conversion.jsp



Step 1: First, you paste your *Arabidopsis* proteins into (A:) on the left-hand side of the page.
Step 2: Select Identifier can be either "ENSEMBL_PROTEIN_ID" or "Not Sure" (at the very bottom of the options it pulls up)
Step 3: You want to click Gene List
Now, on the right-hand side of the page, fill in your reference species (again, mine's *Arabidopsis*) in the "For species:" box. The "Option 1" part there should also already have the

conversion set to “ENTREZ_GENE_ID (Default)” - change it to that if it’s not set up that way already.
Step 4: Click Submit List.

This should give you the list of Entrez IDs that you can now append to your table that has your DE genes, log2FC, and *Arabidopsis* protein labels.

With this table produced, our final piece of work takes us back into RStudio. I have my code for it below:

```
#read in your table with all that information
read.csv("blastptophitsepalearlyupnospliceentrez.csv")->earlyentrezsepalup

#install clusterProfiler
BiocManager::install("clusterProfiler")
library("clusterProfiler")

#install whatever reference database you need for your GO search (this one below is the
Arabidopsis TAIR10 one that I also mentioned in my Phytozome work above)
BiocManager::install("org.At.tair.db")
library("org.At.tair.db")

#perform GO analysis for your DE genes through either of the two ways:

#way 1:
enriched<-enrichGO(gene = earlyentrezsepalup$ENTREZ,
          OrgDb = org.At.tair.db,
          pvalueCutoff =0.05,
          qvalueCutoff =0.05,
          ont="all",
          readable =T)

#results visualized here:
dotplot(enriched, showCategory=20) + ggtitle("dotplot for early sepal glands")

#way 2:
GO_analysis <- enrichGO(gene = earlyentrezsepalup$Protein,
                  OrgDb = org.At.tair.db,  # contains the TAIR/Ensembl id to GO correspondence
for A. thaliana
                  keyType = "TAIR",
                  ont = "ALL",             # either "BP", "CC" or "MF",
                  pAdjustMethod = "BH",
                  pvalueCutoff = 0.05,
                  qvalueCutoff = 0.05,
```

```
                readable = TRUE,
                pool = TRUE)

#results visualized here:
dotplot(GO_analysis, showCategory=30, font.size=12) + ggtitle("Dotplot for Early Sepal
Glands")
dotplot(GO_analysis, showCategory=10, font.size=12) + ggtitle("Dotplot of Top 10 Categories
for Early Sepal Glands")


#You can also perform KEGG metabolic analysis through this package:
keggenriched<-enrichKEGG(gene = earlyentrezsepalup$Protein,
            organism = "ath",
            pvalueCutoff =0.05,
            qvalueCutoff =0.05)

#visualized here:
dotplot(keggenriched, showCategory=30, font.size=12) + ggtitle("KEGG for Early Sepal
Glands")
```

From here, follow-up work on your DE genes to verify their expression and the localization of their expression would be good next steps. The only part of that that I got to was using reciprocal blasting to help verify the identity of my putative homologs.