10/31

Talking to Dr. Yi, we decided to try again, but on a databse that has just human, zebra finch, and chicken proteins.

10/27

faSomeRecords to get the transcripts from 1150 that were found in hetero. Use those to blastx against the tetrapod database (blastx\_Tetrapod.sh). Unfortuantely this timed out at 10 hours. It only got through 187 trascripts, with 87 hits, so extrapolating it would take 54 days to finish and have ~11,000 hits.

10/24

Using the database alias did not work. It said some referenced files could not be found (?).

So I tried faSomeRecords on nr.fasta using the tetrapod gi list, and no records were returned. I suspect because nr.fasta lists many GIs for each sequence?

So I tried to download a fasta of all tetrapod sequences in the same way I got the GI list. This returned only 200k sequences rather than 4 million.

So I wanted to use idfetch to get the sequences, but it depends on shared libraries and I couldn't figure out how to get it going on biocluster. My best guess was using LD\_LIBRARY\_PATH, but that did not fix it.

So I used ebot to retrieve the sequences and it got all 4 million.

When I tried to use makeblastdb on the fasta of tetrapod sequences, it failed because there are some characters it didn't recognize in the sequences (I think).

So I made cleanFasta.py to find sequences with bad characters, and gave the GIs to faSomeRecords -exclude to make a clean fasta file.

But when I try to use that for makeblastdb, I got an error that it didn't know what makeblastdb is. So I sent a support request into pace. Also job submission is often timing out.

Instead I used an interactive biocluster session and then I could run makeblastdb. But I noticed warnings about characters and figured out it was because sometimes there wasn't a new line before >gi, so it looks like part of the preceding sequence. To fix that I used

sed -i "s/\(.\)>gi/\1\n>gi/" cleanTetra.fasta

Making the database, there were still some instances of

Ignoring sequence 'lcl|3552404' as it has no sequence data

but when I looked I didn't see why, so I just let it happen.

We got 4164902 entries in the database from 4165264 sequences in the fasta, so we lost 362 sequences. Not too bad.

10/20

Blast plan:

~~Pull out one column of noSharedPolymorphisms (1150 – column 1)~~

~~Use faSomeRecords to make a search file~~

Make a local copy of nr, using only Tetrapoda, following

http://www.biostars.org/p/6528/

Use blastx against this Tetrapoda database

10/17

Talk to Dr. Yi:  
~~Reread Jim's manuscript~~

Blast search from our 25,000 matches into 2/2m transcriptome

-to weed out shared polymorphisms

-so throw out any pair where at least one is not found in heterozygote, as this indicates a shared polymorphism (98%+)

Do a protein blast search to make sure the transcripts are coding. Also eliminate any pair which codes for a bacteria protein.

Don't worry about using fancy translation, just do simple.

Notes:

Got the ~25k candidates using faSomeRecords

To make the file input to faSomeRecords, used

cat sortedDiffs | awk -F ',' '{print $1}' > 1128\_candidates

Read in both results of searching into 1131, walk through sortedDiffs, keep results with column 1 in 1150 and column 2 in 1128.

23543 entries where each transcript is found in heterozygote

10/14

I decided against merging alignments because maybe there are caused by alternate splicing.

I filtered down to results where the target and query sequence are both no more than 5% longer than the alignment length using transcriptLength.py

26279 matches have query and target sequence no more than 5% longer than alignment length.

I removed all matches where the alignment spans both query and target sequences and had 100% identity. I left matches with 100% alignment identity but with different lengths for query and target sequence. These might be indels, or they might just be something weird with the de novo transcriptome assembly.

25573 matches with percent identity < 100

Sorted with

sort -t "," -g -k6,6 noIdenticals > sortedDiffs

TODO: Look at my links about going from transcript to protein, save those out to a new set of files, and then use biopython (?) to go through the pairs to find nonsynonymous changes.

-Learn which software

-Connect it to my data

-Learn biopython

-Do the biopython work

10/12

I noticed there were duplicate entries in the file of reciprocal matches, so I removed duplicates with:

sort -t "," -u -k1,1 reciprocal > distinctReciprocals

Distinct reciprocals:

138787

Next step is to merge alignments, because there might be more than one alignment between two transcripts. To do that I'll use BRM Parser. <http://www.bioinfo.de/isb/2007/07/0027/>

9/20

Added length into blast results.

blastn -query 1150\_TS.fasta -db db\_1128 -out 1150\_search\_1128\_v2 -outfmt '10 qseqid sseqid qlen slen length pident' -max\_target\_seqs 1

blastn -query 1128\_SWS.fasta -db db\_1150 -out 1128\_search\_1150\_v2 -outfmt '10 qseqid sseqid qlen slen length pident' -max\_target\_seqs 1

Found reciprocal matches with reciprocal.py:

~~Reciprocal matches~~

~~142590 reciprocal matches~~

**Second attempt:**

9/19

blastn -query 1128\_SWS.fasta -db db\_1150 -out 1128\_search\_1150\_v2 -outfmt '10 qseqid sseqid pident' -max\_target\_seqs 1

blastn -query 1150\_TS.fasta -db db\_1128 -out 1150\_search\_1128\_v2 -outfmt '10 qseqid sseqid pident' -max\_target\_seqs 1

To submit all jobs in the folder:

ls blast\*.sh | xargs -n1 msub

Stats:

Memory ~1.1GB  
Time ~1 hour

Results:

1128\_search\_1150\_v1 16MB 272,733 lines (out of 298,236)

1150\_search\_1128\_v1 16MB 266,553 lines (out of 343,164)

Further processing:

~~Rerun with more output options, like lengths~~

~~Find reciprocal~~

~~Filter to make sure alignment length is similar to transcript length~~

~~Remove all entries with 100% identity~~

~~Sort for lowest % identity to get the top X to look at~~

For top X, make sure it is protein coding RNA and look for non-synonymous changes

Blast these against other species

**First attempt:**

9/18

blastn -query 1128\_SWS.fasta -db db\_1150 -out 1128\_search\_1150 -outfmt 10

blastn -query 1150\_TS.fasta -db db\_1128 -out 1150\_search\_1128 -outfmt 10

Stats:

Memory ~1.2GB  
Time ~5 hours

Results:

1128\_search\_1150\_v1 543MB

1150\_search\_1128\_v1 542MB

Improvements:

Make separate sh files for each search so they can run in parallel.

Include only one result per sequence, since we are just looking to match the best.

Remove some extra info from the output format.

**Starting Point**

Transcriptome data assembled by Trinity. Three females:

1128\_SWS - Super white ZAL2M/ZAL2M

1131\_WS - White ZAL2M/ZAL2

1150\_TS - Tan ZAL2 /ZAL2

There was a potential name swap between 1131 and 1150.

in a readme from Jim: "note that 1128=1128 but 1131=1150 and 1150=1131. ie, some kind of switch occurred, probably when samples run or prepped" I have used the names that should match reality. i.e. 1128=1128, 1131=1131, 1150=1150

The files contain sequences and claim to match them to chromosome, but Dr. Yi says don't trust that information.

For now we are comparing Super White to Tan, i.e. 1128 to 1150.

How many sequences in each file?

grep ">" $1 | wc -l

(Added this as a command in ~/bin/countseqs)

1128\_SWS has 298,236 sequences

1150\_TS has 343,164 sequences

**Overview**

Find reciprocal best matches using blast search both ways across the two data sets. Look for nonsynomous mutations where alignment length is the same as the query and the result length. Blast those results against all species, looking for ones related to behavior.

**Blast**

To use blast, I added it to my biocluster account by adding this line to .pacemodules:

module load ncbi\_blast/2.2.28

To use it, I mostly followed these directions:

<http://www.blaststation.com/freestuff/en/howtoNCBIBlastMac.html>

To make databases for blast use, see makeDB.sh

**Future Plans**

compare outputs, find sequences that match, examine differences

Other analysis:

Thomas talk 9/20

Alternate splicing

Examine the transcripts with no blast result