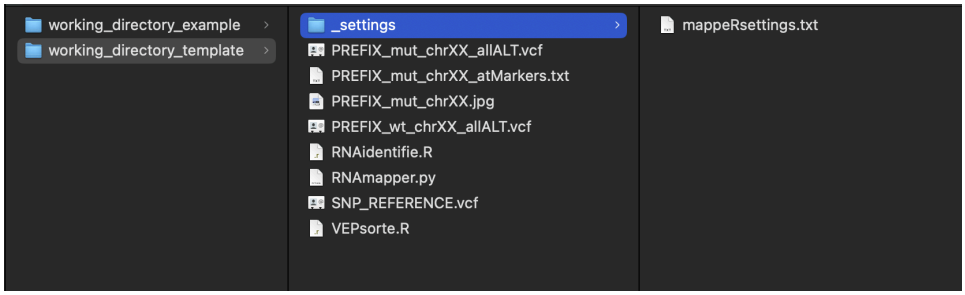
# Step 3: RNAIdenfie.R and VEPSorte.R

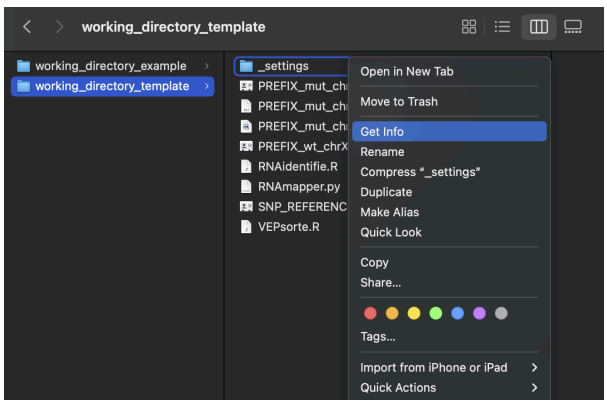
For step 3 of the pipeline, we will be using RNAIdentifie.R to extract and filter variants from the linked region, followed by ENSEMBL's Variant Effect Predictor (VEP) along with the VEPSorte.R script to predict which variants are most likely to impact phenotype.

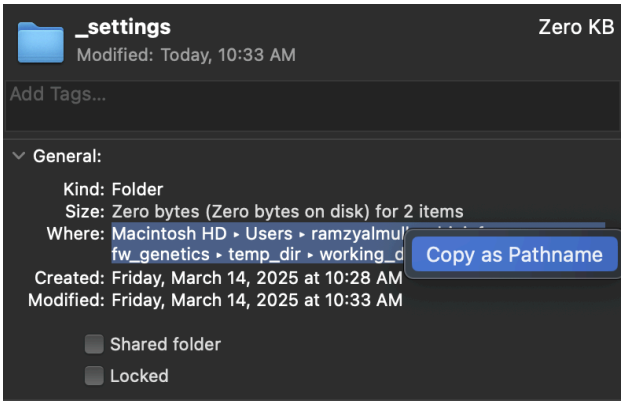
At this point, your directory should look like this:




With a set of "chrXX" files for each chromosome/contig from the previous steps.

Before we begin, make sure your terminal is pointed to the right directory. First, right click on anything inside your top "working directory" folder, then select "get info". The "where" section of the info window does not look like something you can click on, but you can! Right click on that and you can copy the path to this folder.




Now type "cd" into the terminal followed by the pathname you just copied:

cd /path/to/working_directory

Now we are ready to get started with the final step of the pipeline!

## RNAidentifie.R

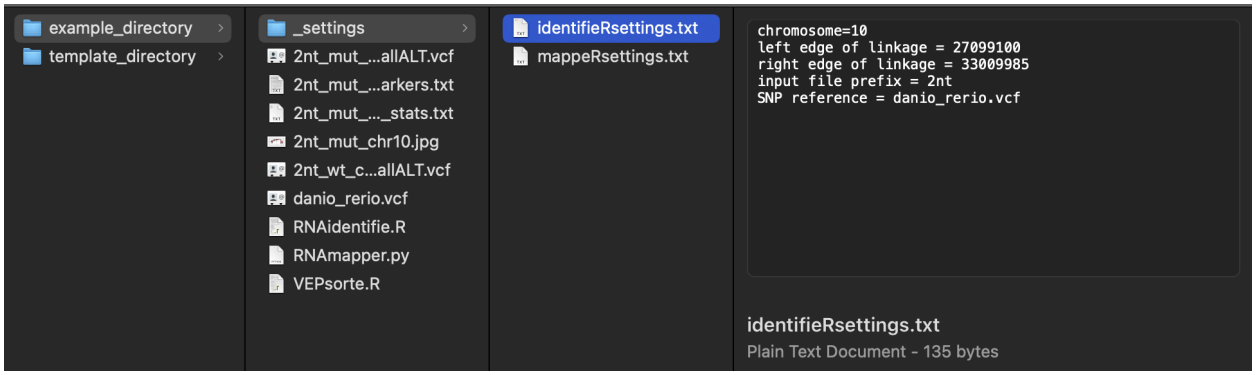
Before you begin, you must first create an identifieRsettings.txt file in your _settings/ directory. This contains important information required to run RNAIdentifie.R, such as the location of your linked region. It should contain the following lines of text:

chromosome=X
left edge of linkage=Y
right edge of linkage=Z
input file prefix=PREFIX
SNP reference=B

This can be done in the terminal using the following commands:

```
cd _settings                                                    # move to _settings folder
echo “chromosome=X” > identifieRsettings.txt                    # create settings file
echo “left edge of linkage=Y” >> identifieRsettings.txt         # append lines to settings file
echo “right edge of linkage=Z” >> identifieRsettings.txt
echo “input file prefix=PREFIX” >> identifieRsettings.txt
echo “SNP reference=B” >> identifieRsettings.txt
cd ..                                                           # move back up to working directory
```

Here is an example where we are interested in a linked region on chr10 of the “2nt” dataset:




You are now ready to run RNAidentifie.R! Based on the outputs from RNAmapper.py, RNAidentifie.R will extract all the variant calls from inside the linked region and filter them against your reference of known SNPs. Just enter the following command:

Rscript --vanilla RNAidentifie.R

This will create a new subfolder, _RNAidentifeR/, which contains the outputs from this script, including a VEPinput.vcf file to be used in the next step.

## Variant Effect Predictor (VEP) & VEPsorte.R

VEP is a powerful tool for predicting how a specific variant will affect the animal. Please visit this page on their website if you are interested to read more about how this is accomplished.

We will be using the VEPinput.vcf file created by RNAidentifie.R as our input file. Note that vep can take over 3 hours to run in database mode! If you anticipate running VEP multiple (e.g., if you want to explore different settings for RNAmapper and/or RNAidentifieR) then it is highly recommended you create a cache of the genome you want to analyze (see VEP's website for more information on this). Their directions are somewhat confusing and out-dated, so I have provided some instructions in a supplemental section to this documentation.

To run VEP in database mode, simply enter the following commands:

```
cd _RNAidentifieR
vep --database -i VEPinput.vcf -o VEPoutput.txt --species Zebrafish --check_existing
```

Once VEP has finished running, we will run the VEPsorte.R script, which sorts VEP's output based on its predicted severity.

```
cd ..
Rscript --vanilla VEPsorte.R
```

This produces two new files, _SNPcandidates.txt and _INDELcandidates.txt, located in the _RNAidendifieR/ folder.

## Interpreting Results

Congratulations! You have successfully completed the RNAmapper pipeline.

Use excel to open the output tables called _SNPcandidates.txt and _INDELcandidates.txt found in the _RNAidentifier directory.

This file will allow you to identify candidate SNP mutations that affect genes within the linkage region you defined. The list will be sorted with SNPs that create nonsense mutations at the top, followed by missense mutations, and followed by other potential changes. Generally, the number of nonsense and missense changes is small and these are of most interest. But there are many changes identified in the UTRs and the less--conserved regions of the genome, which are included for the user to evaluate. However, changes that affect these other categories are 1) many and 2) hard to know if they are informative. Caution needs to be used in interpreting the list.

**THE LIST OUTPUT BY THE ABOVE PROCESS IS FAR FROM PERFECT.** There are several steps where SNP data was retained that does not represent good candidate mutation data. As one example, when multiple alternative alleles are called in the

mutant data there is no way (within the pipeline) to differentiate the number of calls for ALT call #1 or ALT call #2. Lets say that within the mutant data at position X there were 0 REF (C) calls and 14 ALT calls of which 13 were A, and 1 T. Within the pipeline there is no way to know the number of A or T reads, however if the codon at this site is CAG, then the alternative codons would be AAG and TAG. Thus, the latter codon would come through the pipeline as a stop. However, given that there is only 1 T call, this data is likely read error generated by the Illumina process and must be removed from further consideration. **YOU THE USER MUST REMOVE THIS BY EXAMINING THE TABLE DIRECTLY AND EVALUATING THE OUTPUT FOR QUALITY.**

To evaluate each candidate use IGV ( http://www.broadinstitute.org/igv/). IGV allows for aligned RNA--Seq data to be viewed aligned to the genome. Each position identified as a candidate above must be viewed in IGV – this will allow for errors like, but not limited to, the one above to be identified and removed from further consideration. I recommend loading the mutant and wildtype sibling .bam files into IGV. This allows for each site to be evaluated for quality. Additionally, I load independent wildtype sequencing data to assess for known wildtype alternative alleles at the site – if there are independent wildtype alleles that are the same as that which is becoming homozygous in the mutant data, it is unlikely to cause the phenotype of interest and should be thought of as a very low priority candidate.

**Quality candidates will have ALT calls that approach homozygosity and do not have known wildtype alleles of the same type at the site.**

The table is organized by likely severity of consequence, with SNPs creating/destroying STOPs at the top, followed by missense, followed by many other categories as output by VEP such as UTRs, Upstream, etc. The table consists of many columns, but is basically a modified VEP output file, with the sequencing information attached for easy comparison. Some of the less obvious columns are described below:

<u>The first 14 columns are information on the nature of the SNP</u>
***POS*** – chromosomal position, from RNA--Seq mapping data, and put as the first column to more quickly understand where the potential candidate lies within the linkage region.
***#Uploaded_variation*** – all SNPs identified at this location and sent to VEP. The next column (Allele) is being evaluated in this row, the other alleles are elsewhere in the list and can be accessed by searching for the POS.
***Allele*** – the SNP that is being evaluated in this row
***Gene*** – ENSEMBL's gene ID
***Feature*** – ENSEMBL's transcript ID. Note that often the same position is repeated many times in the table – this is because VEP evaluates each transcript, and reports the effect on each. Thus, if there are three transcripts for gene X, and a nonsense

mutation is created that is in frame for each, all three will be reported. You will notice that the same POS/Gene is repeated three times in such a case, with the transcript varying in each row.
***Feature_type*** – Transcripts, other.
***Consequence*** – effect of the SNP on the feature. STOPs are sorted to the top of this table, followed by NON_SYNONOMOUS (missense) followed by many other categories that could be of interest but are likely not useful. However, they are output for completeness.
***Existing_variation*** – Ensemble compares the uploaded variation against their dbSNP information and marks here if there is known variation. However, it does not report the nature of the variation, but this information can be used to look at ENSEMBL's records and find if the SNP identified in your RNA-Seq is a known variation, and therefore unlikely to be causative.
***Extra*** – miscellaneous extra information from VEP

The next several columns are RNA-Seq information for evaluating the quality of the candidate. These are the same as those output by RNAmapper.py. As discussed above, each position should be further investigated with IGV for quality.

***#CHROM*** – chromosome
***POS*** – chromosomal position
***REF*** – the reference allele at this position
***ALT*** – the alternative allele call at this position
***INFO*** – the critical information on the number of reads called at this position. This is the data that is extracted and used for allele frequency calculations by the RNAmappe.R script.
***Fref*** – The number of reference reads aligned in the forward orientation.
***Rref*** – The number of reference reads aligned in the reverse orientation.
***Falt*** – The number of alternative reads aligned in the forward orientation.
***Ralt*** – The number of alternative reads aligned in the reverse orientation.
***refTot*** – Total number of ref reads.
***altTot*** – Total number of alt reads.
***Tot*** – Total number of reads.
***refRat*** – The ratio (frequency) of ref alleles.
***altRat*** – The ratio (frequency) of alt alleles.
***highAllele*** – The greater of refRat and altRat. This is plotted against chromosomal position as black marks in the individual chromosome graphs.
***highAve*** – An average of 50 neighboring highAllele points, with a step size of 1. This smooths the noise and is plotted in the genome wide graph as well as the red line of the individual chromosome graphs

The next 7 columns are information from the known WT variant used to filter the candidate list, with the columns similar to those described above. This

information is included to quickly evaluate if there is known WT variation at the location. Because of limitations in processing, some WT variation cannot be used as a filter. For example, if there are two WT ALT alleles called in addition to the REF allele (e.g. REF is A, ALT is T,G), then there is no information on the quality of the T or G call, and this cannot be used as a filter – the T or the G may be sequencing error. Again, this information should be evaluated using IGV, but is included here to quickly assess the likelihood of problems.