The main pipeline for the manuscript

1. De novo transcriptome assembly

I created two different assembly; one from C3 homografts and another from C4 homografts.

To do so, I created two sample list file contatining the samples that were used for the assembly construction:

Sample list table for C3 homografts:

Tissue #	Replication	Left	Right
HM.Root	R.C3_C3.R1	78062_1_paired.fastq.gz	78062_2_paired.fastq.gz
HM.Root	R.C3_C3.R2	78064_1_paired.fastq.gz	78064_2_paired.fastq.gz
HM.Root	R.C3_C3.R3	78066_1_paired.fastq.gz	78066_2_paired.fastq.gz
HM.Shoot	S.C3_C3.R1	78086_1_paired.fastq.gz	78086_2_paired.fastq.gz
HM.Shoot	S.C3_C3.R2	78088_1_paired.fastq.gz	78088_2_paired.fastq.gz
HM.Shoot	S.C3_C3.R3	78090_1_paired.fastq.gz	78090_2_paired.fastq.gz

Sample list table for C4 homografts:

Tissue #	Replication	Left	Right
HM.Root	R.C4_C4.R1	78080_1_paired.fastq.gz	78080_2_paired.fastq.gz
HM.Root	R.C4_C4.R2	78082_1_paired.fastq.gz	78082_2_paired.fastq.gz
HM.Root	R.C4_C4.R3	78084_1_paired.fastq.gz	78084_2_paired.fastq.gz
HM.Shoot	S.C4_C4.R1	78104_1_paired.fastq.gz	78104_2_paired.fastq.gz
HM.Shoot	S.C4_C4.R2	78106_1_paired.fastq.gz	78106_2_paired.fastq.gz

Tissue #	Replication	Left	Right
HM.Shoot	S.C4_C4.R3	78108_1_paired.fastq.gz	78108_2_paired.fastq.gz

1.1. Trinity run for De novo transcriptome assembly:

Trinity --seqType fq --samples_file "Homo.C3.sample.txt" --max_memory 40G --CPU 14

Note: I did not know whether the read files were strand specific or not, so that's why I ran it first without --SS_lib_type, and then examine the strand specificity in the next step.

1.2. Examine Strand Specificity of RNA-Seq Reads

Note: I did not know if the reads were sequenced stranded or not.

So, first, I ran Trinity without --SS_lib_type option and then I used the guidence in this link to find if they are stranded or not (https://github.com/trinityrnaseq/trinityrnaseq/wiki/Examine-Strand-Specificity).

First, align your reads back against your Trinity assembly. Bowtie2 works well for this:

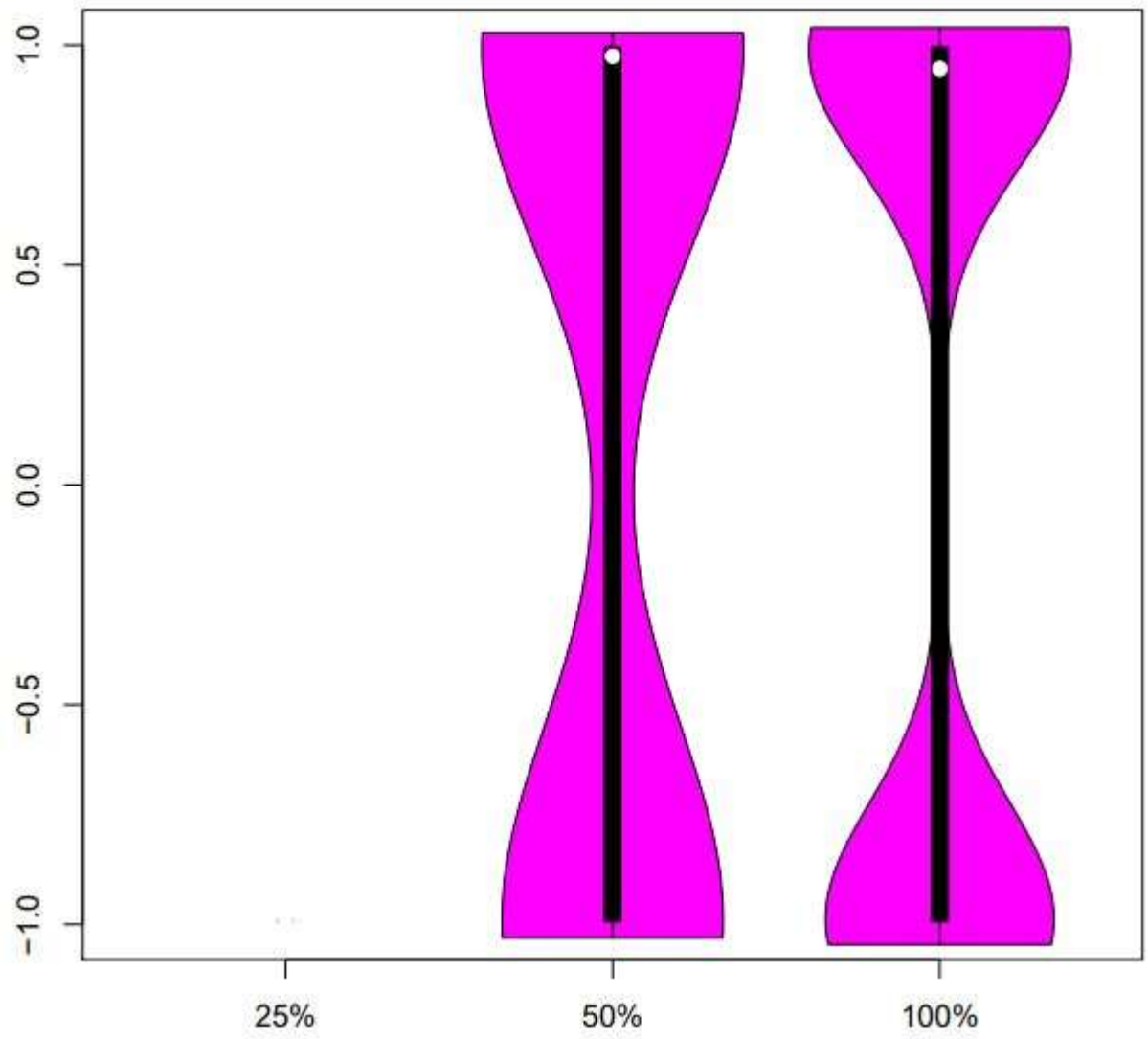
/usr/lib/trinityrnaseq/util/misc/run_bowtie2.pl --target Trinity.fasta --left 78080_1_paired.fastq --right 78080_2_paired.fastq | samtools view -Sb - | samtools sort - -o bowtie2.coordSorted.bam mkdir Strand_specificity_dir mv *.bt2 /mkdir Strand_specificity_dir cd /mkdir Strand_specificity_dir

Then, examine the distribution of strand-specificity - looking at the distribution of orientations for the first read of paired-end fragment reads.

/usr/lib/trinityrnaseq/util/misc/examine_strand_specificity.pl bowtie2.coordSorted.bam So, when I realized that it is stranded, I renamed the trinity output dir and ran the analysis against cd ../ rm trinity_out_dir unstrand_trinity_out_dir

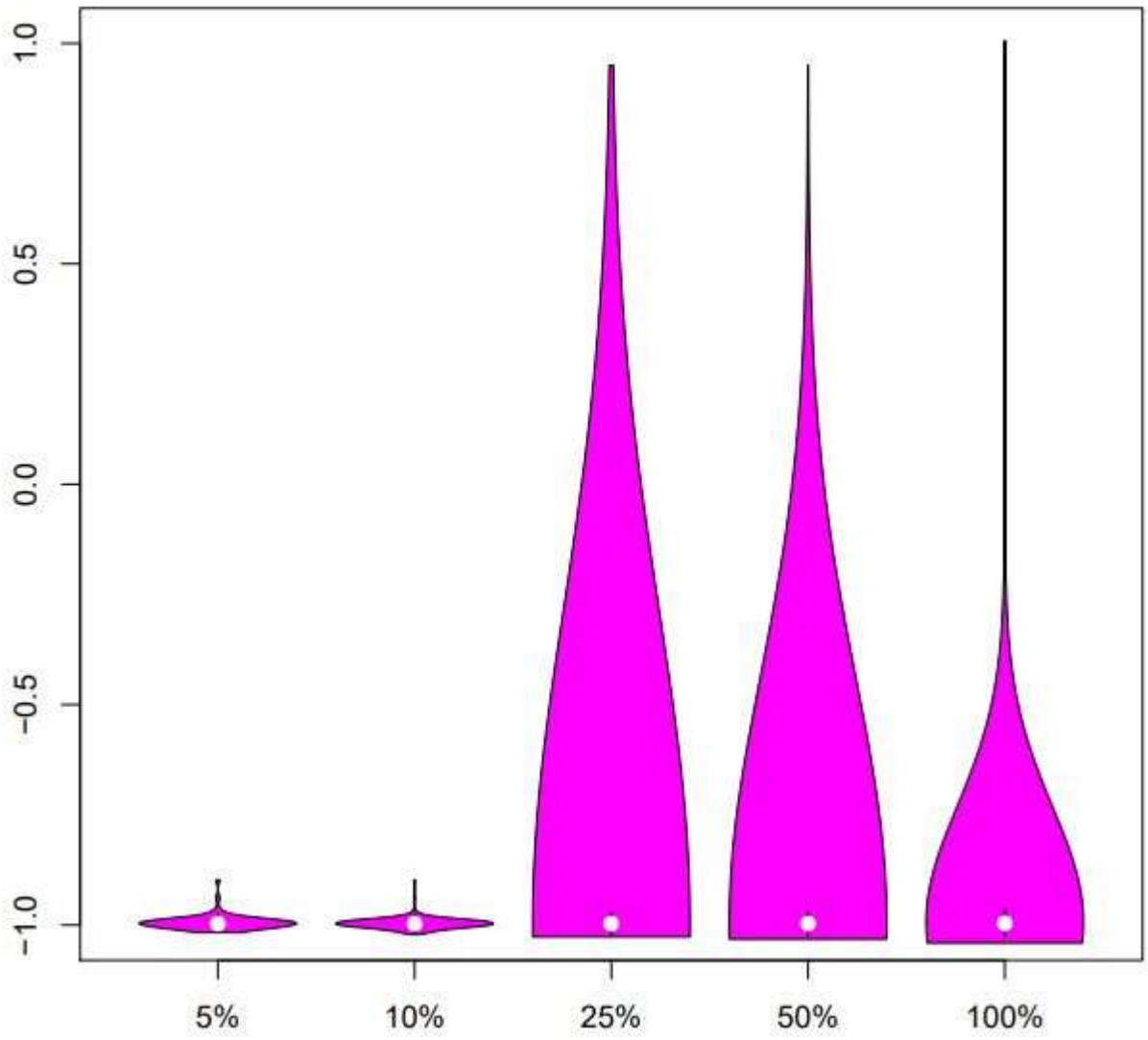
1.2.1. An unstranded example.

This is from my C4 assembly that I made it without --SS_lib_type



1.2.2.Stranded example.

This is from C3 assembly that I made it with --SS_lib_type



1.3. Run Trinity as Stranded reads:

Trinity --seqType fq --samples_file "Homo.C3.sample.txt" --SS_lib_type RF --max_memory 40G --CPU 14

2. Assembly quality assessment

2.1. Assessing the read content of the transcriptome assembly

2.1. RNA seq Read Representation

In order to comprehensively capture read alignments, we run the process below. Bowtie2 is used to align the reads to the transcriptome and then we count the number of proper pairs and improper or orphan read

alignments. Related link: https://github.com/trinityrnaseq/trinityrnaseq/wiki/RNA-Seq-Read-Representation-by-Trinity-Assembly

First, build a bowtie2 index for the transcriptome:

bowtie2-build Trinity.fasta Trinity.fasta

Then perform the alignment to just capture the read alignment statistics.

bowtie2 -p 10 -q --no-unal -k 20 -x Trinity.fasta -1 reads_1.fq -2 reads_2.fq \ 2>align_stats.txt| samtools view -@10 -Sb -o bowtie2.bam

The output from C3 homograft was copied here:

28489039 (100.00%) were paired; of these: 3042513 (10.68%) aligned concordantly 0 times 10653739 (37.40%) aligned concordantly exactly 1 time

14792787 (51.92%) aligned concordantly >1 times

3042513 pairs aligned concordantly 0 times; of these:

377855 (12.42%) aligned discordantly 1 time

2664658 pairs aligned 0 times concordantly or discordantly; of these:

5329316 mates make up the pairs; of these:

1064874 (19.98%) aligned 0 times

1270360 (23.84%) aligned exactly 1 time

2994082 (56.18%) aligned >1 times

98.13% overall alignment rate

Interpration:

Bowtie2 found 3042513 read pairs that it couldn't align concordantly and looked for discordant alignments for them. It could align 377855 of them discordantly, leaving 2664658 still unmapped. It then tried to align each of the reads in those pairs separately (i.e., as singletons), which worked for about 79% of them.

So, I could say 88% (37.40 + 51.92) mapped or 98.13% in overall that is a good percentage and can say the quality of assembly is good.

A typical Trinity transcriptome assembly will have the vast majority of all reads mapping back to the assembly, and ~70-80% of the mapped fragments found mapped as proper pairs (yielding concordant alignments 1 or more times to the reconstructed transcriptome).

2.2. Transcriptome Contig Nx and ExN50 stats

Below we describe Trinity toolkit utilities for computing contig Nx statistics (eg. the contig N50 value), in addition to a modification of the Nx statistic that takes into consideration transcript expression (read support) data, which we call the ExN50 statistic.

The 'Gene' Contig Nx Statistic

Based on the lengths of the assembled transcriptome contigs, we can compute the conventional Nx length statistic, such that at least x% of the assembled transcript nucleotides are found in contigs that are at least of Nx length. The traditional method is computing N50, such that at least half of all assembled bases are in transcript contigs of at least the N50 length value.

The following script in the Trinity toolkit will compute these values for you like so:

TRINITY_HOME/util/TrinityStats.pl Trinity.fasta

2.2.1 Nx stat for C3:

#

Counts of transcripts, etc.

#

Total trinity 'genes': 47158

Total trinity transcripts: 88253

Percent GC: 40.69

#

Stats based on ALL transcript contigs:

#

Contig N10: 3620

Contig N20: 2849

Contig N30: 2374

Contig N40: 2037

Contig N50: 1752

Median contig length: 911

Average contig: 1179.64

Total assembled bases: 104106958

#

Stats based on ONLY LONGEST ISOFORM per 'GENE':

#

Contig N10: 3493

Contig N20: 2715

Contig N30: 2258

Contig N40: 1929

Contig N50: 1644

Median contig length: 627

Average contig: 988.83

Total assembled bases: 46631287

2.2.2 Nx stat for C4:

#

Counts of transcripts, etc.

#

Total trinity 'genes': 46304

Total trinity transcripts: 83692

Percent GC: 40.37

#

Stats based on ALL transcript contigs:

#

Contig N10: 3792

Contig N20: 2998

Contig N30: 2515

Contig N40: 2159

Contig N50: 1864

Median contig length: 956

Average contig: 1236.91

Total assembled bases: 103519264

#

Stats based on ONLY LONGEST ISOFORM per 'GENE':

#

Contig N10: 3615

Contig N20: 2825

Contig N30: 2353

Contig N40: 1999

Contig N50: 1704

Median contig length: 620

Average contig: 1005.51

Total assembled bases: 46558986

3.Trinity Transcript Quantification

My goal to perform this step:

This step will give me the expression level as count, FPKM, and TPM. I will run this for both homo root and shoot against their assembly for both C4 and C3. Then I will run it for both hetero root and shoot against their assemblies for both C4 and C3. And then I can run DEG analysis for both root and shoot that how transcriptome profile was changed in root or shoot when they are homografted and heterografted.

Tissue#	Condition	Assembly
Root	C3.C3	C3.assembly
Root	C3.C4	C3.assembly
Root	C4.C4	C4.assembly
Root	C4.C3	C4.assembly
Shoot	C3.C3	C3.assembly
Shoot	C4.C3	C3. assembly
Shoot	C4.C4	C4.assembly
Shoot	C3.C4	C4.assembly

3.1. Estimating Transcript Abundance

There are now several methods available for estimating transcript abundance in a genome-free manner, and these include alignment-based methods (aligning reads to the transcript assembly) and alignment-free methods (typically examining k-mer abundances in the reads and in the resulting assemblies).

In Trinity, we provide direct support for running the alignment-based quantification methods RSEM, as well as the ultra-fast alignment-free method kallisto and 'wicked-fast' salmon.

The Trinity software does not come pre-packaged with any of these software tools, so be sure to download and install any that you wish to use. The tools should be available via your PATH setting (so, typing 'which kallisto'on the linux command line returns the path to where the tool is installed on your system).

If you have multiple RNA-Seq data sets that you want to compare (eg. different tissues sampled from a single organism), be sure to generate a single Trinity assembly and to then run the abundance estimation separately for each of your samples.

3.2. RSEM method

Just prepare the reference for alignment and abundance estimation:

/usr/lib/trinityrnaseq/util/align_and_estimate_abundance.pl --transcripts Trinity.fasta --est_method RSEM --aln_method bowtie --trinity_mode --prep_reference

Run the alignment and abundance estimation (assumes reference has already been prepped, errors-out if prepped reference not located.)

/usr/lib/trinityrnaseq/util/align_and_estimate_abundance.pl --transcripts Trinity.fasta --gene_trans_map Trinity.fasta.gene_trans_map --seqType fq --samples_file "" --SS_lib_type RF --est_method RSEM --aln_method bowtie --trinity_mode --output_dir rsem_outdir

Note: I used --samples_file option so, do not need run the --left and --right options and run the command for each condition. It will store output in the folder with condition name.

If you have strand-specific data, be sure to include the '--SS_lib_type' parameter.

It is useful to first run 'align_and_estimate_abundance.pl' to only prep your reference database for alignment, using '--prep_reference', and then subsequently running it on each of your sets of reads in parallel to obtain sample-specific abundance estimates.

PART 2: FINDING TRANSMITTED RNAs

In this section I will perform different mapping to find the RNAs that are transmitted in heterografts: