Starling-May18
Projects/Katarina Stuart/KStuart.Starling-Aug18/At1_Genome/Annotation/2023-01-19.Transcriptome

PDF Version generated by

Katarina Stuart (z5188231@ad.unsw.edu.au)

on

Aug 23, 2023 @09:36 AM NZST

Table of Contents

2023-01-19.Transcriptome

Katarina Stuart (z5188231@ad.unsw.edu.au) - Jun 08, 2023, 4:40 PM GMT+12

Transcriptome

data is in:

```
/nesi/project/uoa02613/RNAseq/
```

Read processing and mapping

setting up the necessary files and dir structure

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome/
mkdir read_processing

#creates file, one line per fastq pairs, R1 in col 1, R2 in col 2
ls -lh /nesi/project/uoa02613/RNAseq/KIN5180_20180803-88947859/FASTQ_Generation_2018-08-03_20_02_09Z-113587657/*/* | awk -F " " '{print $9}' | tr "\n" " " | sed 's/ /\n/2; P; D' > RNAsamples_array_list.txt

wc -l RNAsamples_array_list.txt #12 for array script header
```

Index the genome

Run the trimming and mapping

```
#!/bin/bash -e

#SBATCH --job-name=2023_01_26.transcriptome_index.sl
#SBATCH --account=uoa02613
#SBATCH --time=00-02:00:00
#SBATCH --mem=4GB
#SBATCH --output=%x_%j.errout
#SBATCH --mail-user=katarina.stuart@auckland.ac.nz
#SBATCH --mail-type=ALL
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=2
#SBATCH --profile task

module load HISAT2/2.2.1-gimpi-2022a

cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome/read_processing

GENOME=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/curation/step4_scaffolding/ragtag_atris_synteny/renamed/AcTris_vAus2.0.fasta
hisat2-build $GENOME $(basename $GENOME .fasta)
```

Run the trimming and mapping

```
#!/bin/bash -e

#SBATCH --job-name=2023_01_26.transcriptome_mapping.sl
#SBATCH --account=uoa02613
#SBATCH --time=00-12:00:00
```

```
#SBATCH --mem=4GB
#SBATCH --output=%x_%j.errout
#SBATCH --mail-user=katarina.stuart@auckland.ac.nz
#SBATCH --mail-type=ALL
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=2
#SBATCH --profile task
#SBATCH --array=1-12

module load HISAT2/2.2.1-gimpi-2022a
module load SAMtools/1.15.1-GCC-11.3.0
module load fastp/0.23.2-GCC-11.3.0

cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome/read_processing

R1=$(sed "${SLURM_ARRAY_TASK_ID}q;d" ../RNAsamples_array_list.txt | awk -F " " '{print $1}')
R2=$(sed "${SLURM_ARRAY_TASK_ID}q;d" ../RNAsamples_array_list.txt | awk -F " " '{print $2}')

#read trimming and qc
fastp -i $R1 -o $(basename $R1 .fastq.gz).trimmed.fastq.gz -I $R2 -O $(basename $R2 .fastq.gz).trimmed.fastq.gz -z 4

#map the reads
GENOME=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/curation/step4_scaffolding/ragtag_atris_synteny/renamed/AcTris_vAus2.0.fasta

hisat2 -x $(basename $GENOME .fasta) -1 $(basename $R1 .fastq.gz).trimmed.fastq.gz -2 $(basename $R2 .fastq.gz).trimmed.fastq.gz -S $(basename $R1
.fastq.gz).sam --phred33 --novel-splicesite-outfile $(basename $R1 .fastq.gz).junctions --rna-strandness RF --dta -t

samtools sort -o $(basename $R1 .fastq.gz).sorted.bam $(basename $R1 .fastq.gz).sam
```

Stringtie

https://github.com/gpertea/stringtie

https://ccb.jhu.edu/software/stringtie/

Download stringtie v2 (v1 on nesi, and I want to use the new one)

```
cd /nesi/nobackup/uoa02613/kstuart_projects/programs
git clone https://github.com/gpertea/stringtie
    cd stringtie
   make release
```

Run stringtie

On masking the genome: run single tissue only first and see how it is
'In order to predict genes accurately in a novel genome, the genome should be masked for repeats. This will avoid the prediction of false positive gene structures in repetitive and low complexitiy regions. Repeat masking is also essential for mapping RNA-Seq data to a genome with some tools (other RNA-Seq mappers, such as HISAT2, ignore masking information). " - Breaker3 manual

```
#!/bin/bash -e

#SBATCH --job-name=2023_01_23.transcriptome_assembly.sl
#SBATCH --account=uoa02613
#SBATCH --time=00-04:00:00
#SBATCH --mem=4GB
#SBATCH --output=%x_%j.errout
#SBATCH --mail-user=katarina.stuart@auckland.ac.nz
#SBATCH --mail-type=ALL
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=2
#SBATCH --profile task

module load SAMtools/1.15.1-GCC-11.3.0

cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome

#transcritpome by tissue
for tissue in Liver Heart Test
do
samtools merge -o $tissue.sorted.bam read_processing/*${tissue}*.sorted.bam
/nesi/nobackup/uoa02613/kstuart_projects/programs/stringtie/stringtie -o $tissue.gtf $tissue.sorted.bam
done

#whole transcriptome
/nesi/nobackup/uoa02613/kstuart_projects/programs/stringtie/stringtie --merge *.gtf -o all_tissues.gtf
```

Analysis

Comparing tissue specific transcripts

https://github.com/gpertea/gffcompare

https://ccb.jhu.edu/software/stringtie/gffcompare.shtml

downloading gffcompare

```
cd /nesi/nobackup/uoa02613/kstuart_projects/programs
git clone https://github.com/gpertea/gffcompare
  cd gffcompare
  make release
```

Compare the transcript outputs across tissue types using gffcompare tracking

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome

ls -lh *.gtf | awk -F " " '{print $9}' > gffcompare_list.txt

/nesi/nobackup/uoa02613/kstuart_projects/programs/gffcompare/gffcompare -o transcript.tracking  -i gffcompare_list.txt
```

Plot using an upset plot

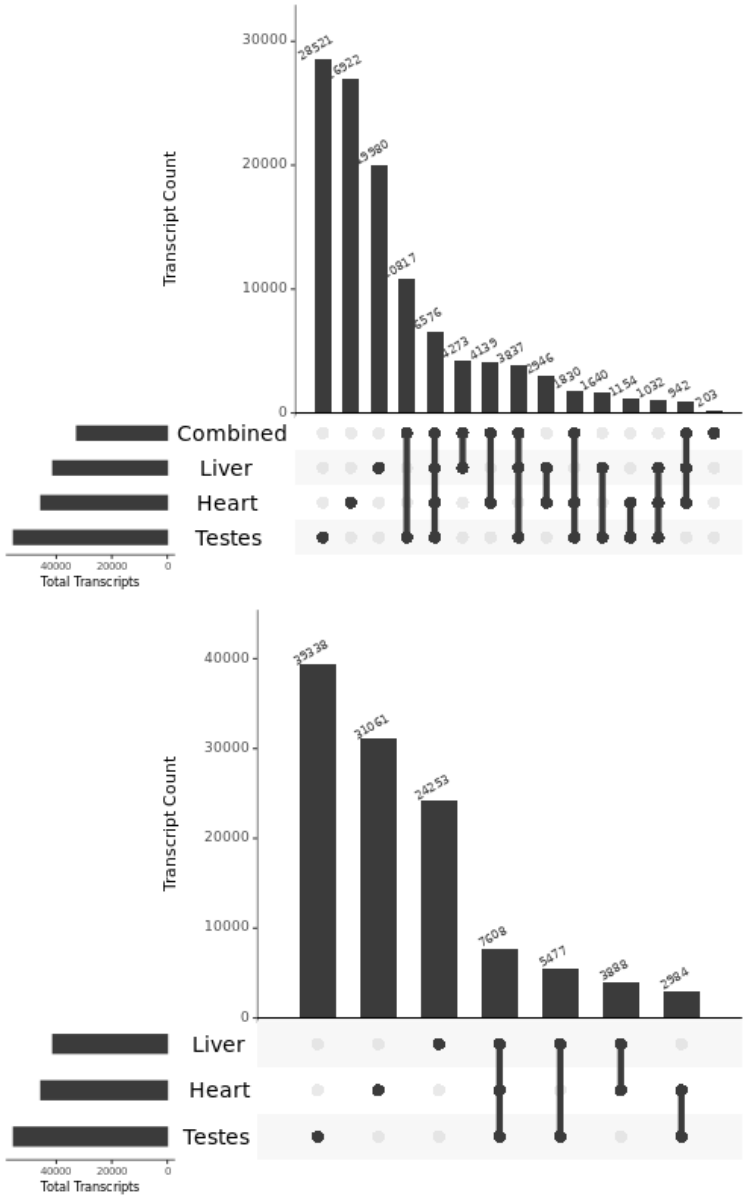
https://cran.r-project.org/web/packages/UpSetR/vignettes/basic.usage.html

prepare input data for plot

```
#make one file per sample
awk '$5 != "-" {print $0}' transcript.tracking.tracking | cut -f1 > all_tissues.upset
awk '$6 != "-" {print $0}' transcript.tracking.tracking | cut -f1 > Heart.upset
awk '$7 != "-" {print $0}' transcript.tracking.tracking | cut -f1 > Liver.upset
awk '$8 != "-" {print $0}' transcript.tracking.tracking | cut -f1 > Test.upset
```

and plot the data

```
module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome")
file1<-scan("all_tissues.upset", what = "", quiet=TRUE)
file2<-scan("Heart.upset", what = "", quiet=TRUE)
file3<-scan("Liver.upset", what = "", quiet=TRUE)
file4<-scan("Test.upset", what = "", quiet=TRUE)  #total transcripts

all_files <- list(Combined = file1, Liver = file2, Heart = file3, Testes = file4)
all_files <- list(Liver = file2, Heart = file3, Testes = file4)

#install.packages("UpSetR")
library(UpSetR)

png("Transcript_upsetplot.png", width=500, height=400)
upset(fromList(all_files), order.by = "freq", empty.intersections = "on", point.size = 3.5, line.size = 2, mainbar.y.label = "Transcript Count", sets.x.label = "Total
Transcripts", text.scale = c(1.3, 1.3, 1, 1, 2, 1.3), number.angles = 30 )
dev.off()

pdf("Transcript_upsetplot.pdf")
upset(fromList(all_files), order.by = "freq", empty.intersections = "on", point.size = 3.5, line.size = 2, mainbar.y.label = "Transcript Count", sets.x.label = "Total
Transcripts", text.scale = c(1.3, 1.3, 1, 1, 2, 1.3), number.angles = 30 )
dev.off()
```



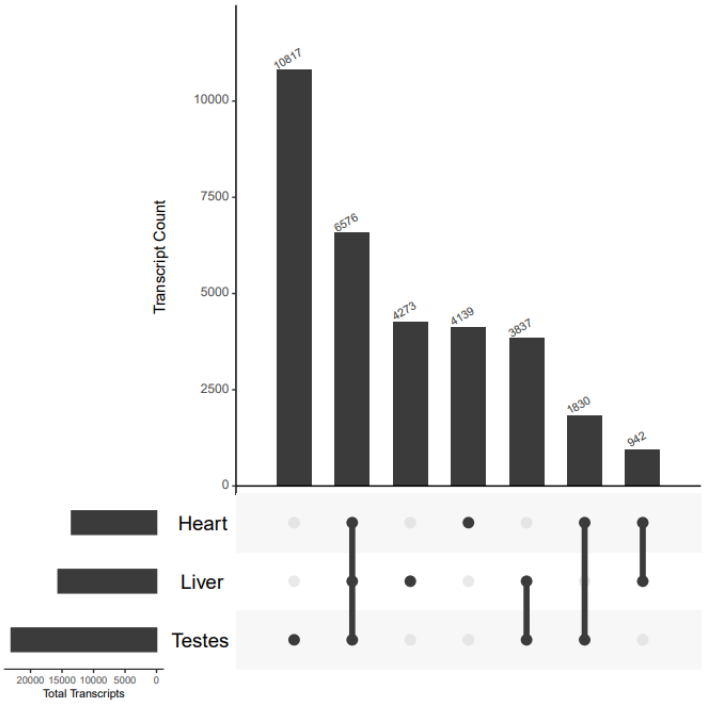
modifying the input data so it is just those transcripts contributing to the combined transcript

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome
awk '$5 != "-" {print $0}' transcript.tracking.tracking > transcript_combined.tracking.tracking
awk '$6 != "-" {print $0}' transcript_combined.tracking.tracking | cut -f1 > Heart_overlap.upset
awk '$7 != "-" {print $0}' transcript_combined.tracking.tracking | cut -f1 > Liver_overlap.upset
awk '$8 != "-" {print $0}' transcript_combined.tracking.tracking | cut -f1 > Test_overlap.upset
```

and plot the data

```
module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome")
file2<-scan("Heart_overlap.upset", what = "", quiet=TRUE)
file3<-scan("Liver_overlap.upset", what = "", quiet=TRUE)
file4<-scan("Test_overlap.upset", what = "", quiet=TRUE)

all_files <- list(Liver = file2, Heart = file3, Testes = file4)
```

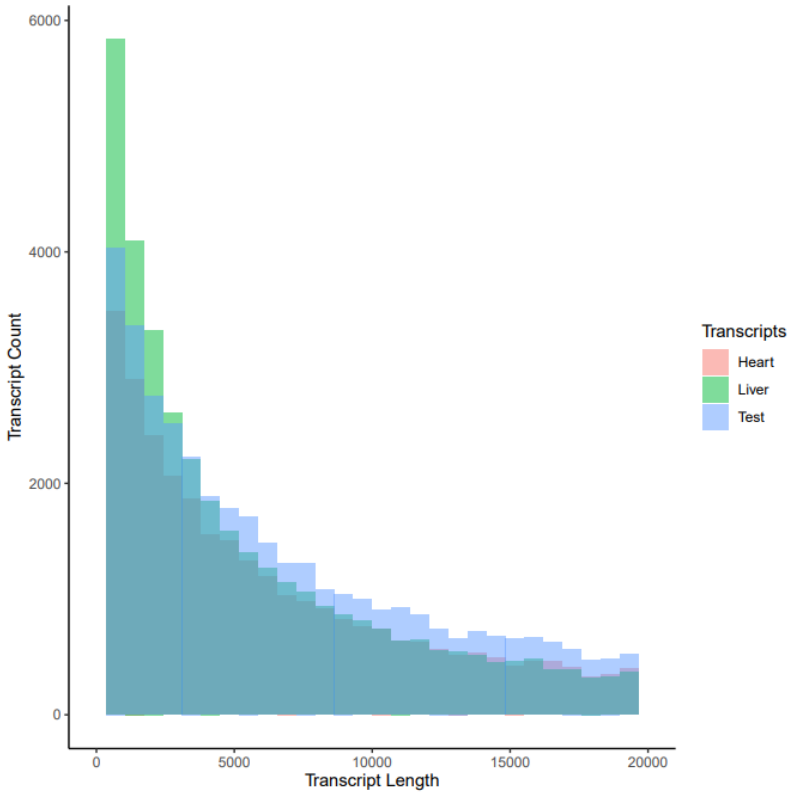
```
#install.packages("UpSetR")
library(UpSetR)

pdf("Transcript_combine_upsetplot.pdf")
upset(fromList(all_files), order.by = "freq", empty.intersections = "on", point.size = 3.5, line.size = 2, mainbar.y.label = "Transcript Count", sets.x.label = "Total Transcripts", text.scale = c(1.3, 1.3, 1, 1, 2, 1.3), number.angles = 30 )
dev.off()
```



Size distribution of transcripts

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome

awk '$3 == "transcript"' Heart.gtf | awk '{print "Heart",'\t',$5-$4}' > Heart_transcript_lengths.txt
awk '$3 == "transcript"' Liver.gtf | awk '{print "Liver",'\t', $5-$4}' > Liver_transcript_lengths.txt
awk '$3 == "transcript"' Test.gtf | awk '{print "Test",'\t', $5-$4}' > Test_transcript_lengths.txt

cat Heart_transcript_lengths.txt Liver_transcript_lengths.txt Test_transcript_lengths.txt > All_transcript_lengths.txt

module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome")
library("ggplot2")
library("dplyr")

histo <- read.table(file="All_transcript_lengths.txt", header=FALSE, sep=" ")

heart <- histo %>% filter(V1=="Heart")
test <- histo %>% filter(V1=="Test")
liver <- histo %>% filter(V1=="Liver")

pdf("Transcript_histo.pdf")
ggplot(histo, aes(x=V3, fill=V1)) + geom_histogram(position="identity", alpha=0.5) + xlim(0,20000) + theme_classic() + ylab('Transcript Count') + xlab('Transcript Length') + labs(fill='Transcripts')
dev.off()

pdf("Transcript_histo_heart.pdf", width=6, height=2)
```

```
ggplot(heart, aes(x=V3, fill=V1)) + geom_histogram(position="identity") + xlim(0,20000) + theme_classic() + ylab('Transcript Count') + xlab('Transcript Length') +
labs(fill='Transcripts') + scale_fill_manual(values=c("#E69F00"))
dev.off()

pdf("Transcript_histo_liver.pdf", width=6, height=2)
ggplot(liver, aes(x=V3, fill=V1)) + geom_histogram(position="identity") + xlim(0,20000) + theme_classic() + ylab('Transcript Count') + xlab('Transcript Length')
+ labs(fill='Transcripts') + scale_fill_manual(values=c("#0e3e87"))
dev.off()

pdf("Transcript_histo_test.pdf", width=6, height=2)
ggplot(test, aes(x=V3, fill=V1)) + geom_histogram(position="identity") + xlim(0,20000) + theme_classic() + ylab('Transcript Count') + xlab('Transcript Length')
+ labs(fill='Transcripts') + scale_fill_manual(values=c("#7d4c29"))
dev.off()
```



BUSCO

```
#!/bin/bash -e

#SBATCH --job-name=2023_04_19.busco_transcriptome.sl
#SBATCH --account=uoa02613
#SBATCH --time=00-12:00:00
#SBATCH --mem=15GB
#SBATCH --output=%x_%j.errout
#SBATCH --mail-user=katarina.stuart@auckland.ac.nz
#SBATCH --mail-type=ALL
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=16
#SBATCH --profile task

module purge
module load BUSCO/5.3.2-gimkl-2020a

cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/transcriptome
```

```
#module load BEDTools/2.30.0-GCC-11.3.0
#bedtools getfasta -fi /nesi/nobackup/uoa02613/kstuart_projects/programs/breaker3/genome2.fa -bed <(awk '$3=="transcript"' all_tissues.gtf ) -fo all_tissues.fa
#module load SeqKit/2.2.0
#seqkit rmdup -s < all_tissues.fa > all_tissues_unique.fa

module load BEDTools/2.30.0-GCC-11.3.0
module load SeqKit/2.2.0

for TISSUE in Heart Liver Test
do
bedtools getfasta -fi /nesi/nobackup/uoa02613/kstuart_projects/programs/breaker3/genome2.fa -bed <(awk '$3=="transcript"' ${TISSUE}.gtf ) -fo ${TISSUE}.fa
seqkit rmdup -s < $TISSUE.fa > ${TISSUE}_unique.fa
done

busco -i Liver_unique.fa -o busco_Liver -m transcriptome -l aves_odb10 -c 16 -f
busco -i Heart_unique.fa -o busco_Heart -m transcriptome -l aves_odb10 -c 16 -f
busco -i Test_unique.fa -o busco_Test -m transcriptome -l aves_odb10 -c 16 -f
```