PURDUE UNIVERSITY

RNA-SEQ DOCUMENTATION

Shuvra Kanti Nath

June 4, 2015

Summary: This document shows the steps in details to generate read count data from gtf files for a genome.
Machine: Carter
Softwares needed: Flux Simulator, samtools, bowtie, tophat
Data in : /scratch/carter/n/naths
Steps for Human Genome processing:

Step 1(Get Genome Data)

First download the chromFa.tar.gz from http://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/ and untar it.

Step 2(Run Flux Simulator)

Run:
/homes/naths/flux-simulator-1.2.1/bin/flux-simulator -p hg19.par .
If successful, a .fasta file will be generated. See the hg19.par bellow.

```
## File locations
REF_FILE_NAME   chr1_refseq_sub.gtf
GEN_DIR         genomes/


## Expression
NB_MOLECULES    20000
TSS_MEAN NaN
POLYA_SCALE     NaN
POLYA_SHAPE     NaN
EXPRESSION_K -0.9

## Fragmentation
FRAG_SUBSTRATE  RNA
FRAG_METHOD     UR

## RT parameters
RTRANSCRIPTION YES
RT_MOTIF       default
RT_PRIMER      RH
RT_LOSSLESS    YES
RT_MIN         500
RT_MAX         5500

## PCR / Filtering
PCR_DISTRIBUTION  default
FILTERING         YES
SIZE_SAMPLING     AC
SIZE_DISTRIBUTION N(200,25)

# Amplifcation
GC_MEAN NaN

# Sequencing
```

```
READ_NUMBER      20000
READ_LENGTH      50
PAIRED_END       YES
FASTA            YES
UNIQUE_IDS YES
```

Step 3(Download gtf file)

Download .gtf file from https://genome.ucsc.edu/cgi-bin/hgTables. Use(clade: mammal, genome:human, assembly: hg19, track: refseq genes, table: refgene, output format : gtf file format). The file should look like Figure **??**.

```
chr1  hg19_refGene  start_codon 67000042  67000044  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67000042  67000051  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  66999825  67000051  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67091530  67091593  0.000000  +  2  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67091530  67091593  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67098753  67098777  0.000000  +  1  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67098753  67098777  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67101627  67101698  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67101627  67101698  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67105460  67105516  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67105460  67105516  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67108493  67108547  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67108493  67108547  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67109227  67109402  0.000000  +  2  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67109227  67109402  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67126196  67126207  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67126196  67126207  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67133213  67133224  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67133213  67133224  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67136678  67136702  0.000000  +  0  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  exon  67136678  67136702  0.000000  +  .  gene_id "NM_032291"; transcript_id "NM_032291";
chr1  hg19_refGene  CDS 67137627  67137678  0.000000  +  2  gene_id "NM_032291"; transcript_id "NM_032291";
```

Figure 1: gtf file

Step 4(Choose a subset from the gtf)

The downloaded gtf file is very big and will take a very long processing time. So, we want a subset of gtf file corresponding to few genes and their isoforms. From https://genome.ucsc.edu/cgi-bin/hgTables, select group: all tables, table: refFlat, output format: all fields from selected table. The selected table will look like: Figure **??**.

```
#geneName   name    chrom   strand  txStart txEnd   cdsStart    cdsEnd  exonCount   exonStarts  exonEnds
DDX11L1 NR_046018   chr1    +   11873   14409   14409   14409   3   11873,12612,13220,  12227,12721,14409,
WASH7P  NR_024540   chr1    -   14361   29370   29370   29370   11  14361,14969,15795,16606,16857,17232,17605,17914,18267,24737,29320,
14829,15038,15947,16765,17055,17368,17742,18061,18366,24891,29370,
MIR6859-4   NR_128720   chr1    -   17368   17436   17436   17436   1   17368,  17436,
MIR6859-3   NR_107063   chr1    -   17368   17436   17436   17436   1   17368,  17436,
MIR6859-1   NR_106918   chr1    -   17368   17436   17436   17436   1   17368,  17436,
MIR6859-2   NR_107062   chr1    -   17368   17436   17436   17436   1   17368,  17436,
MIR1302-10  NR_036267   chr1    +   30365   30503   30503   30503   1   30365,  30503,
MIR1302-2   NR_036051   chr1    +   30365   30503   30503   30503   1   30365,  30503,
MIR1302-9   NR_036266   chr1    +   30365   30503   30503   30503   1   30365,  30503,
MIR1302-11  NR_036268   chr1    +   30365   30503   30503   30503   1   30365,  30503,
FAM138A NR_026818   chr1    -   34610   36081   36081   36081   3   34610,35276,35720,  35174,35481,36081,
FAM138F NR_026820   chr1    -   34610   36081   36081   36081   3   34610,35276,35720,  35174,35481,36081,
OR4F5   NM_001005484    chr1    +   69090   70008   69090   70008   1   69090,  70008,
LOC729737   NR_039983   chr1    -   134772  140566  140566  140566  3   134772,139789,140074,   139696,139847,140566,
LOC100133331    NR_028327   chr1    +   323891  328581  328581  328581  4   323891,324287,324438,327035,    324060,324345,326938,328581,
LOC100132062    NR_028325   chr1    +   323891  328581  328581  328581  3   323891,324287,324438,   324060,324345,328581,
LOC100132287    NR_028322   chr1    +   323891  328581  328581  328581  3   323891,324287,324438,   324060,324345,328581,
OR4F3   NM_001005224    chr1    +   367658  368597  367658  368597  1   367658, 368597,
OR4F16  NM_001005277    chr1    +   367658  368597  367658  368597  1   367658, 368597,
OR4F29  NM_001005221    chr1    +   367658  368597  367658  368597  1   367658, 368597,
LOC101928626    NR_125957   chr1    -   562759  564389  564389  564389  3   562759,563340,564298,   563203,563603,564389,
```

Figure 2: gene and isoform table

#geneName and name show the name of gene and corresponding isoform names. Now you can choose any genes and the corresponding isoforms that you want to process. In a list.txt file write down the isoform names like:

```
NM_021170
NM_001142467
```

```
NR_047526
```

To get a subest of gtf file corresponding to this isoforms, run:

```
grep -f list.txt hg19.gtf > subset.gtf
```

Example:
From the genes of chr1: LINC01128 , LOC100130417, LOC100133331, PERM1, HES4
I chose the isoforms:

```
NM_021170,
NM_001142467,
NR_047526,
NR_047525,
NR_047524,
NR_047523,
NR_047521,
NR_047519,
NR_122045,
NR_026874,
NR_028327,
NM_001291366,
NM_001291367,
NR_028327
```

Step 5 (Download Bowtie indices)

Download pre-built bowtie indices from http://bowtie-bio.sourceforge.net/manual.shtml.
Now run :

```
./make_hg19.sh
```

If it does not work, then run : bowtie2-build chr1.fa chr17.fa hg19
This step will generate 4 files: hg19.1.bt2, hg19.2.bt2, hg19.3.bt2, hg19.4.bt2
.

Step 6 (Run Tophat)

Load tophat:

module use /apps/group/bioinformatics/modules
module load tophat
module load bowtie2
Run tophat:

```
tophat hg19 hg19.fasta (if single end read)
tophat hg19 hg19_1.fasta hg19_2.fasta (if paired-end read)
```

If successful, this will generate accepted_hits.bam in tophat_out folder.

***If you want to simulate single end reads, you need to specify "PARIED_END NO" in the .par file while running flux-simulator. If you had used paired-end reads, then .fasta generated from flux-simulator needed to be seperated into 2 files.
Command:

```
grep 'c.*/1' -A 1 hg19.fasta | sed '/--/d' > hg19_1.fasta
grep 'c.*/2' -A 1 hg19.fasta | sed '/--/d' > hg19_2.fasta
```

Step 7 (Get the sam file)

Run:

```
module load samtools
samtools view -f 67 accepted_hits.bam | awk '$3=="chr1" {print $1,$2,$4,$6,$8,$9}' > sam
```

You can use use flag 67 or 131. To know about the details about the flags go here: http://broadinstitute.github.io/picard/explain-flags.html. The sam file(this was from paired-end) should look like : Figure **??**.

```
chr1:661139-665731C:NR_028327_dup1:535:4273:4083:4210 339 134913 50M 134835 -128
chr5:180750507-180755196W:NR_028327:162:4273:4081:4248 339 134915 50M 134797 -168
chr1:661139-665731C:NR_028327_dup1:414:4273:4079:4232 339 134917 50M 134813 -154
chr1:661139-665731C:NR_028327_dup1:211:4273:4075:4228 83 134921 50M 134817 -154
chr1:661139-665731C:NR_028327_dup1:168:4273:4074:4237 339 134922 50M 134808 -164
chr1:661139-665731C:NR_028327_dup1:229:4273:4072:4212 83 134924 50M 134833 -141
chr1:661139-665731C:NR_028327_dup1:583:4273:4062:4231 339 134934 50M 134814 -170
chr1:661139-665731C:NR_028327_dup1:172:4273:4058:4210 339 134938 50M 134835 -153
chr1:323892-328581W:NR_028327:97:4273:4056:4209 83 134940 50M 134836 -154
chr1:661139-665731C:NR_028327_dup1:447:4273:4048:4210 339 134948 50M 134835 -163
chr1:661139-665731C:NR_028327_dup1:513:4273:3977:4110 339 135019 50M 134935 -134
chr1:323892-328581W:NR_028327:90:4273:3941:4110 339 135055 50M 134935 -170
chr1:661139-665731C:NR_028327_dup1:605:4273:3648:3791 339 135348 50M 135254 -144
chr5:180750507-180755196W:NR_028327:158:4273:3647:3800 339 135349 50M 135245 -154
chr5:180750507-180755196W:NR_028327:289:4273:3606:3758 339 135390 50M 135287 -153
chr1:661139-665731C:NR_028327_dup1:120:4273:3598:3750 339 135398 50M 135295 -153
chr1:661139-665731C:NR_028327_dup1:302:4273:3582:3750 339 135414 50M 135295 -169
chr1:661139-665731C:NR_028327_dup1:67:4273:3549:3716 339 135447 50M 135329 -168
```

Figure 3: Sam file

Step 8 (Generate read count)

In case of paired-end reads, Use the samtools to filter out the first mate and second mate separately. (All the reads and the mapping information are in the .bam file, but you need to separate them out)

```
 samtools view -f 67 accepted_hits.bam | awk '{print $1,$2,$4,$6,$8,$9}' > sim_11.txt
 samtools view -f 97 accepted_hits.bam | awk '{print $1,$2,$4,$6,$8,$9}' > sim_12.txt

samtools view -f 131 accepted_hits.bam | awk '{print $1,$2,$4,$6,$8,$9}' > sim_21.txt
samtools view -f 145 accepted_hits.bam | awk '{print $1,$2,$4,$6,$8,$9}' > sim_22.txt


cat sim_11.txt sim_12.txt  > sim_1.txt
cat sim_21.txt sim_22.txt > sim_2.txt
```

67, 97, 131 and 145 are different kinds of reads and may not be available in every study.

We have a .R script which takes the sam files sim_1.txt, sim_2.txt as input and generates the read count file.