Lotus Genome v3.0 - Methods

Vikas Gupta and Stig U. Andersen

January 10, 2014

Contents

1. Introduction **2**

2. Gene Annotation **2**
2..1 Repeat Masking . . . 2
2..2 Gene model Generation . . . 2
2..2.1 RNA-seq . . . 2
2..2.2 De novo assembled transcripts . . . 4
2..3 Gene model selection and filtering . . . 4

3. Lotus accession resequencing **4**
3..1 Variant calling and filtering . . . 4
3..2 Analysis of phylogenetic relationships and LD . . . 4
3..3 SNP effect nalysis . . . 4

1. Introduction

This document is the detailed description of methods section for the Lotus genome article. Aim is to be so thourogh that all the steps can repeated without requiring additional information. Path of the files will be added accordingly if still exists. I will try to add the python scripts in a package but if there is any missing, you can always fetch is from the my GitHub `https://github.com/vikas0633/python`.

2. Gene Annotation

Primary idea was use the already available genome annotation pipelines/tools, such as PASA, MAKER, EVM and Inchworm but the annotation from the tools mentioned were not very good and so we used a custom build pipeline developed by me and Stig. I will not mention the commands used for the tools which were not used towards the final output.

2..1 Repeat Masking

RepeatScout Version 1.0.5 and RepeatMasker version open-3.3.0 was used for masking the repetitive regions of the genome. RepeatScout was used to construct denovo library from the lotus genome sequence to facilated accurate detection of novel repeat elements. These repeat library was subsequently used with RepeatMasker to mask the repeat regions.

```
### l value using python
>>> math.ceil(math.log(454435385,4)+1)
16.0

### running build_lmer_table from repeat scout
nice -n 19 build_lmer_table -sequence /u/vgupta/01_genome_annotation/01_genome/
    Ljchr0-6_pseudomol_20120830.scaf.fa -l 16 -freq lmer_Ljchr0-6
    _pseudomol_20120830.scaf.fa

### running repeatscout
nice -n 19 RepeatScout -sequence /u/vgupta/01_genome_annotation/01_genome/Ljchr0-6
    _pseudomol_20120830.scaf.fa -output output_RepeatScout_Ljchr0-6
    _pseudomol_20120830.scaf.fa -freq lmer_Ljchr0-6_pseudomol_20120830.scaf.fa -l
    16
Program duration is 5704.0 sec = 95.1 min = 1.6 hr

### filtering step-1
filter-stage-1.prl output_RepeatScout_Ljchr0-6_pseudomol_20120830.scaf.fa >
    output_filter-stage-1_RepeatScout_Ljchr0-6_pseudomol_20120830.scaf.fa

### running repeat masker
nohup nice -n 19 RepeatMasker -gff -lib output_filter-stage-1_RepeatScout_Ljchr0-6
    _pseudomol_20120830.scaf.fa /u/vgupta/01_genome_annotation/01_genome/Ljchr0-6
    _pseudomol_20120830.scaf.fa &
```

2..2 Gene model Generation

2..2.1 RNA-seq

Four pair-end RNA-seq libraries, two from each MG20 and Gifu were mapped on the genome. We ran TopHat and Cufflinks multiple times to find the best suiting parameters for mapping. TopHat v2.0.4 was used together with Bowtie v0.12.8. Tophat aligns the reads to the genome taking exon-intron boundries

into consideration. Aligned reads were used to create gene models using Cufflinks v2.0.2 and many non-default parameters were used to detect all potential gene models.

```
#!/bin/csh
#PBS -l nodes=1:ppn=16
#PBS -q normal

echo "========= Job started  at `date` =========="
echo 'for only MG20 tophat cufflinks'

### get the tools from rune's directory
source /com/extra/bowtie/0.12.8/load.sh
source /com/extra/tophat/2.0.4/load.sh
source /com/extra/cufflinks/2.0.2/load.sh
source /com/extra/samtools/0.1.18/load.sh

### nodes to be used
cores=15

### data_dir
data_dir="/home/vgupta/01_genome_annotation/02_transcriptomics_data"

### work dir
work_dir="/home/vgupta/01_genome_annotation/11_tophat/04"

### log file
logfile=$work_dir"/20120917.logfile"


### reference genome
ref="/home/vgupta/01_genome_annotation/01_genome/Ljchr0-6_pseudomol_20120830.chlo.
    mito.fa"
index="/home/vgupta/01_genome_annotation/01_genome/Ljchr0-6_pseudomol_20120830.chlo
    .mito.fa"


echo 'indexing the genome' >>$logfile
### make index for the reference sequence
bowtie-build -f $ref $index
echo "indexing is finished" >>$logfile

echo 'processing first sample' >>$logfile

read1=$data_dir"/2010_02_17_Fasteris_MG20_Gifu_transcripts/100128_s_1_1_seq_GHD-1.
    txt",\
$data_dir"/2010_02_17_Fasteris_MG20_Gifu_transcripts/100128_s_2_1_seq_GHD-2.txt",\
$data_dir"/2010_03_22_Fasteris_MG20_Gifu_transcripts/100226_s_7_1_seq_GHD-1.txt",\
$data_dir"/2010_03_22_Fasteris_MG20_Gifu_transcripts/100226_s_8_1_seq_GHD-2.txt"

read2=$data_dir"/2010_02_17_Fasteris_MG20_Gifu_transcripts/100128_s_1_2_seq_GHD-1.
    txt",\
$data_dir"/2010_02_17_Fasteris_MG20_Gifu_transcripts/100128_s_2_2_seq_GHD-2.txt",\
$data_dir"/2010_03_22_Fasteris_MG20_Gifu_transcripts/100226_s_7_2_seq_GHD-1.txt",\
$data_dir"/2010_03_22_Fasteris_MG20_Gifu_transcripts/100226_s_8_2_seq_GHD-2.txt"

mkdir $work_dir"/tophat"

### print the file names
echo "Reference file: "$ref >>$logfile
```

```
echo "read 1:"$read1 >>$logfile
echo "read 2:"$read2 >>$logfile

### run tophat
tophat --bowtie1 --num-threads $cores -I 25000 -o $work_dir"/tophat" $ref $read1
    $read2
echo "tophat is done" >>$logfile

### bam_file
bam="accepted_hits.bam"
sam="accepted_hits.sam"

### convert bam to sam

samtools view $work_dir"/"$bam > $work_dir"/"$sam

### run cufflink

cufflinks --pre-mrna-fraction 0.5 --small-anchor-fraction 0.01 --min-frags-per-
    transfrag 5 --overhang-tolerance 20 --max-bundle-length 10000000 --min-intron-
    length 20 --trim-3-dropoff-frac 0.01 --max-multiread-fraction 0.99 --no-
    effective-length-correction --no-length-correction --multi-read-correct --upper
    -quartile-norm  --total-hits-norm --max-mle-iterations 10000  --max-intron-
    length 50000 --no-update-check -p $cores -o $work_dir $work_dir"/"$sam
echo "cufflinks is done" >>$logfile
```

2..2.2 De novo assembled transcripts

2..3 Gene model selection and filtering

3. Lotus accession resequencing

3..1 Variant calling and filtering

3..2 Analysis of phylogenetic relationships and LD

3..3 SNP effect nalysis