Practical II – RNAseq mapping and EdgeR

Isheng Jason Tsai

Introduction to NGS Data and Analysis
Lecture 12

Practical outline

1. Call SNPs

2. Mapping of RNAseq reads
3. Analyse using EdgeR
4. Visualise them in artemis

5. Visualise the variations on bedtools

Install bcftools

Install bcftools http://www.htslib.org/download/

Binary available! --->

Get vcf file

 # Get VCF file
 # Bam file is from last week
 # May take a long time ; so let's just call SNP for the 1st Mb of PNOK.scaff0001.C
samtools mpileup -r PNOK.scaff0001.C:1-1000000 -ugf ref.fa A42_sorted.bam | bcftools call -vmO v -o A42.vcf

What does VCF file look like?

Install hisat2

Install hisat2 https://ccb.jhu.edu/software/hisat2/index.shtml

Binary available! --->

Input files

Reference file same as last week
Again, fastq files:

Paired end reads in pair fastq (_1 and _2) files
Two conditions: fruiting body and fungal mat
Each condition with two replicates (Rep1 and Rep2)
An annotation file in gtf format (ref.gtf)

```
-rw-rw-r-- 1 ijt ijt 152M Jun  1 19:33 fruitRep1_1.fq.gz
-rw-rw-r-- 1 ijt ijt 154M Jun  1 19:33 fruitRep1_2.fq.gz
-rw-rw-r-- 1 ijt ijt 152M Jun  1 19:33 fruitRep2_1.fq.gz
-rw-rw-r-- 1 ijt ijt 154M Jun  1 19:33 fruitRep2_2.fq.gz
-rw-rw-r-- 1 ijt ijt 120M Jun  1 19:33 fungalRep1_1.fq.gz
-rw-rw-r-- 1 ijt ijt 123M Jun  1 19:33 fungalRep1_2.fq.gz
-rw-rw-r-- 1 ijt ijt 119M Jun  1 19:33 fungalRep2_1.fq.gz
-rw-rw-r-- 1 ijt ijt 120M Jun  1 19:33 fungalRep2_2.fq.gz
```

Hisat2

https://ccb.jhu.edu/software/hisat2/manual.shtml

 # You need reference file (ref.fa),
 # Paired end fastqs (A42_1.fq F42_2.fq)

 # Build the database for hisat2
hisat2-build ref.fa ref

 # Map reference
hisat2 -x ref -1 fruitRep1_1.fq.gz -2 fruitRep1_2.fq.gz -S fruitRep1.sam

 # For those with laptop/server with multiple cores (much faster)
hisat2 -p 4 -x ref -1 fruitRep1_1.fq.gz -2 fruitRep1_2.fq.gz -S fruitRep1.sam

 # Can you map the other three samples using the same command with slight modifications?

 # Name the other three samples output as fungalRep2.sam fruitRep1.sam fruitRep2.sam

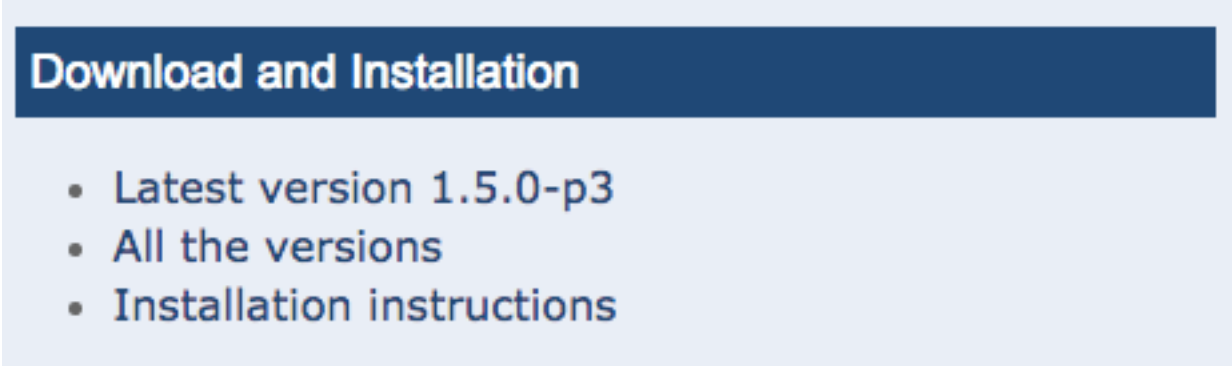
Hisat2 example output

```
2651074 reads; of these:
  2651074 (100.00%) were paired; of these:
    29700 (1.12%) aligned concordantly 0 times
    2207199 (83.26%) aligned concordantly exactly 1 time
    414175 (15.62%) aligned concordantly >1 times
    ----
    29700 pairs aligned concordantly 0 times; of these:
      10446 (35.17%) aligned discordantly 1 time
    ----
    19254 pairs aligned 0 times concordantly or discordantly; of these:
      38508 mates make up the pairs; of these:
        1767 (4.59%) aligned 0 times
        1200 (3.12%) aligned exactly 1 time
        35541 (92.30%) aligned >1 times
99.97% overall alignment rate
```

Install subread package

Install subread http://subread.sourceforge.net/

Binary available! --->



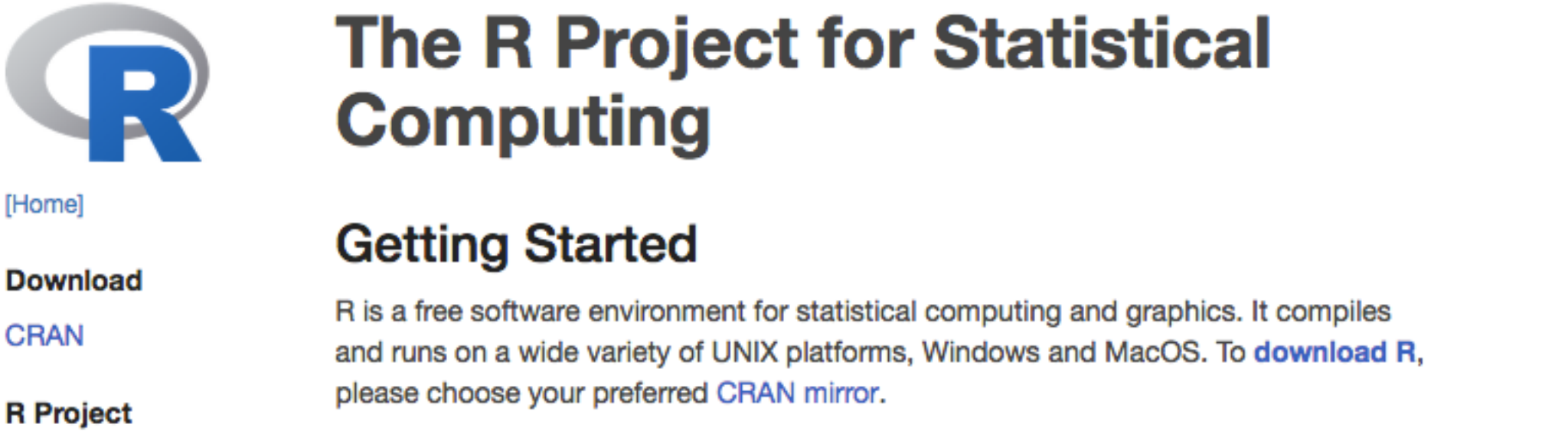
featureCounts

 # Generate a matrix file for edgeR
featureCounts -p -s 2 -t exon -g gene_id -a ref.gtf -o counts.txt fruitRep1.sam fruitRep2.sam fungalRep1.sam fungalRep2.sam

Q: Was the standard output (text appears on screen) informative about how good the mapping is?

Install R and EdgeR package

https://www.r-project.org/



Type the following to install essential packages:

```
source("https://bioconductor.org/biocLite.R")
biocLite("edgeR")
biocLite("locfit")
biocLite("ggplot2")
biocLite("RColorBrewer")
```

EdgeR manual (a good software always keep updated)

https://bioconductor.org/packages/devel/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf

edgeR: differential expression analysis of digital gene expression data

User's Guide

Yunshun Chen, Davis McCarthy,
Matthew Ritchie, Mark Robinson, Gordon K. Smyth

First edition 17 September 2008

Last revised 20 April 2016

Our data frame **df** looks like this, just like excel but it's now easier to manipulate

head(df)

	fruitRep1.sam	fruitRep2.sam	fungalRep1.sam	fungalRep2.sam
PNOK_0000100	0	0	0	0
PNOK_0000200	0	0	0	0
PNOK_0000300	0	0	0	0
PNOK_0000400	0	0	0	0
PNOK_0000500	212	267	37	29
PNOK_0000600	287	448	28	36
PNOK_0000700	1	0	1	0
PNOK_0000800	917	1362	296	365
PNOK_0000900	17	40	2	5
PNOK_0001000	54	75	7	10
PNOK_0001100	2465	4025	1864	1712
PNOK_0001200	1848	1849	7024	6678
PNOK_0001300	1897	1465	985	1311
PNOK_0001400	9	50	160	201
PNOK_0001500	23	66	426	310
PNOK_0001600	1018	1768	43	40

df[c("PNOK_0001100"),]

df[16,1]

df[,2]
df$fruitRep2.sam

Our data frame **df** looks like this, just like excel but it's now easier to manipulate

Try the commands below!

head(df)

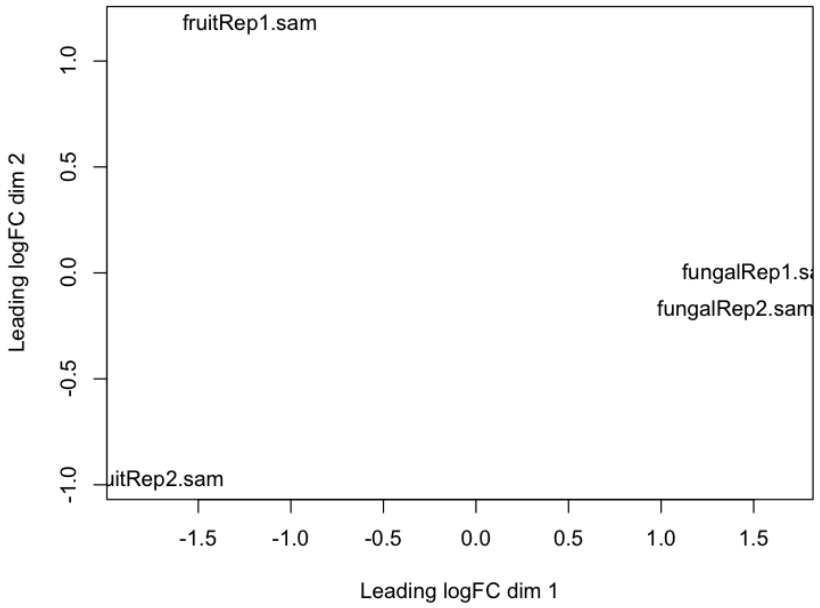
	fruitRep1.sam	fruitRep2.sam	fungalRep1.sam	fungalRep2.sam
PNOK_0000100	0	0	0	0
PNOK_0000200	0	0	0	0
PNOK_0000300	0	0	0	0
PNOK_0000400	0	0	0	0
PNOK_0000500	212	267	37	29
PNOK_0000600	287	448	28	36
PNOK_0000700	1	0	1	0
PNOK_0000800	917	1362	296	365
PNOK_0000900	17	40	2	5
PNOK_0001000	54	75	7	10
PNOK_0001100	2465	4025	1864	1712
PNOK_0001200	1848	1849	7024	6678
PNOK_0001300	1897	1465	985	1311
PNOK_0001400	9	50	160	201
PNOK_0001500	23	66	426	310
PNOK_0001600	1018	1768	43	40

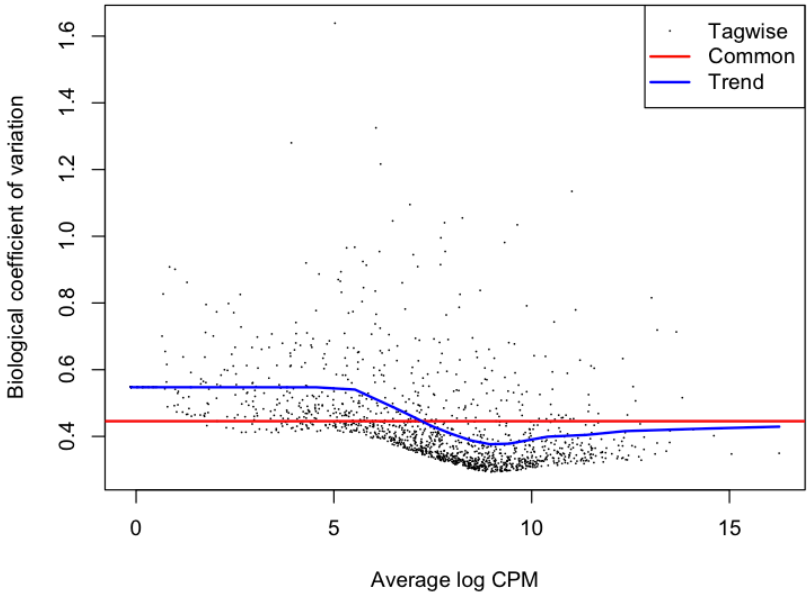
df[c("PNOK_0001100"),]

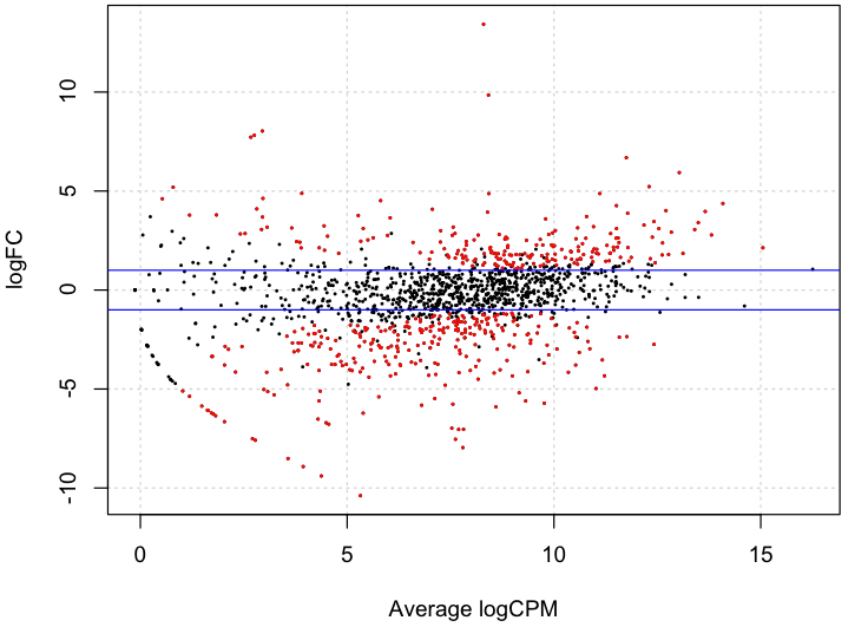
df[16,1]

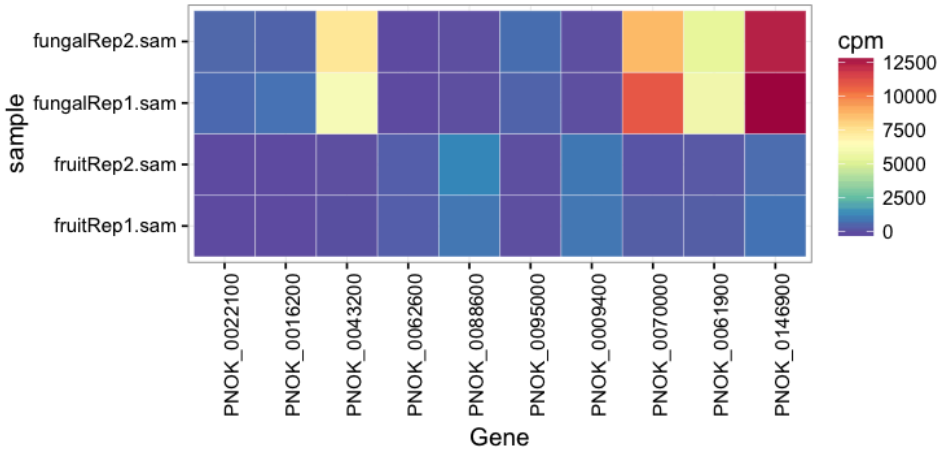
df[,2]
df$fruitRep2.sam

EdgeR plot produced









Visualise them in artemis

1. For each of the four sam files
    a. convert them into bam
    b. sort them and index them

2. Load into artemis
    What do the mapping look like?
    How is it different to genomic DNA mapping?

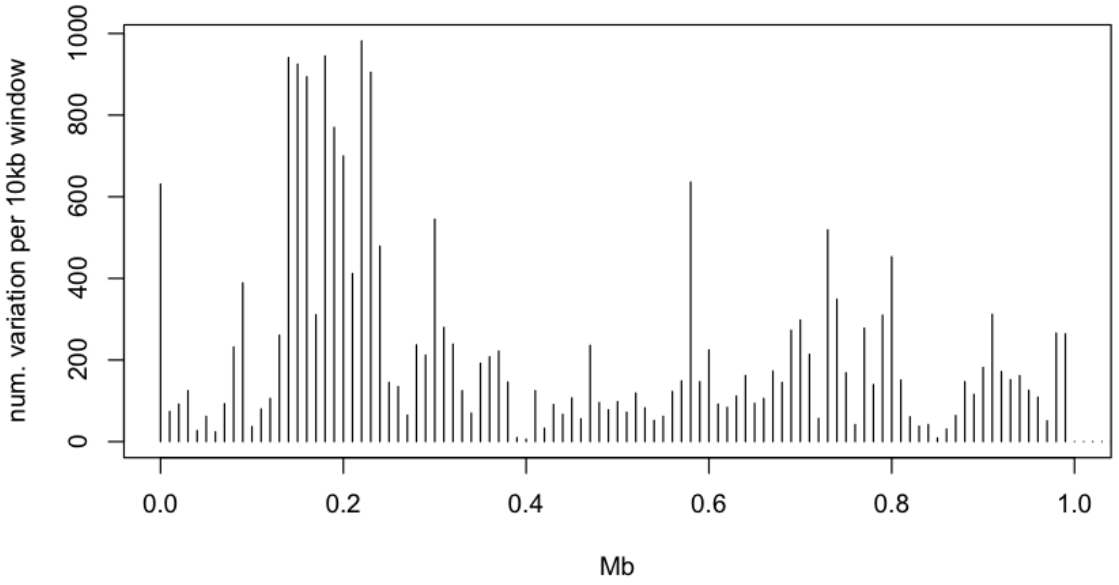
Visualise the number of SNPs per 10kb window

1. Install bedtools (http://bedtools.readthedocs.io/en/latest/)

 # Create a bed file of 10kb window
bedtools makewindows -g ref.fa.fai -w 10000 > ref.fa.bed

 # Do a bed file intersect to check to bin the SNPs in these 10kb windows
 # A42.vcf made in slide 2
bedtools intersect -c -b A42.vcf -a ref.fa.bed > ref.fa.A42.bed

R script to load the bar plot




```
x <- read.table("~/Desktop/ref.fa.A42.bed",header=F)
names(x) <- c("Chr","win_start","win_end","SNPs")
head(x)
```

```
hist(x$SNPs)
hist(x$SNPs,breaks=100)
plot(x$win_start, x$SNPs,type="l")
plot(x$win_start, x$SNPs,type="l", xlim=c(0,1000000))
plot(x$win_start, x$SNPs,type="h", xlim=c(0,1000000))
plot(x$win_start, x$SNPs,type="h", xlim=c(0,1000000),xlab="bp",ylab="num. variation per 10kb window")
plot(x$win_start/1000000, x$SNPs,type="h", xlim=c(0,1),xlab="Mb",ylab="num. variation per 10kb window")
```