

RNAseq workshop

SERMSAWAT TUNLAYA-ANUKIT

MARCH 26, 2014

Outline

- Introduction : workflow of mapping RNAseq
- Step 1 Map sequence into genome by Tophat
- Step 2 count read in each gene by coverageBed
- Step 3 compile the multiple rawcount files into one table by perl
- Step 4 identify differential expressed genes by edgeR
- Question and answer

Workflow of mapping RNASeq

Map sequence into genome by Tophat

Input:fastq

Output:BAM file



Count read in each gene by coverageBed

Input:BAM file

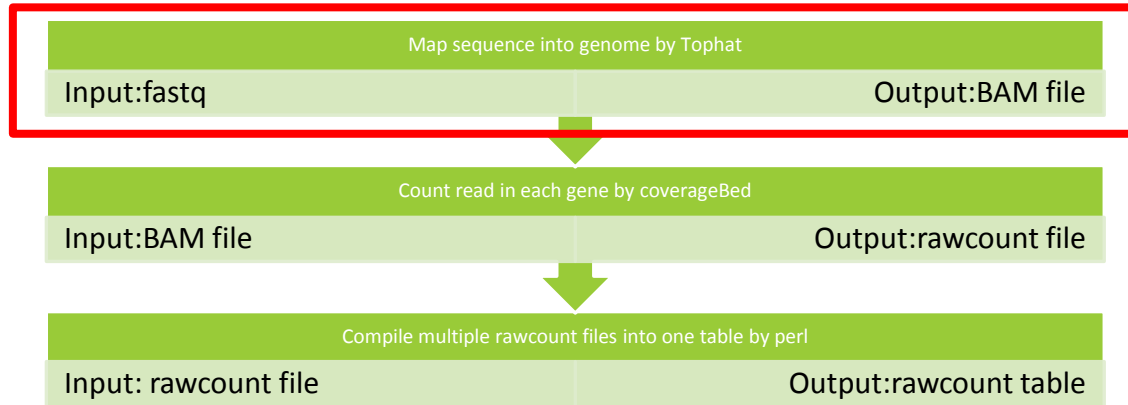
Output:rawcount file



Compile multiple rawcount files into one table by perl

Input: rawcount file

Output:rawcount table



Step 1

MAP SEQUENCE INTO GENOME BY TOPHAT

Edit file : tophat2.pl (for Poplar version2)

```
use strict;

# input files

my @fastqfiles = glob ("/media/Archive1/fastq/tissue_PO/*.fastq");
my $seqindex = "/genomes/ptrichocarpa/indexV2.2/bowtie2-index/Ptrichocarpa_156.f
my $GFF = "/genomes/ptrichocarpa/sequencesV2.2/Ptrichocarpa_156_gene_exons.gff3"

# output directory

my $outdir = "/media/Archive1/BAM/v2.2/tissue_PO/";
mkdir $outdir;

# Process each file

foreach my $qfile (@fastqfiles){
    my @aa = split /\//,$qfile;

    my $filename = pop @aa;
    my @bb = split /\./,$filename;
    my $addir = shift @bb;
    my $outputdir=$outdir.$addir;

    print "\nprocesseing $qfile\n";
    print "tophat -p 7 -G $GFF -o $outputdir $seqindex $qfile\n";

# Using 7 processors

    system "tophat -p 7 -G $GFF -o $outputdir $seqindex $qfile";
}
}
```

Change the location of fastq files

Change the location of result BAM files

Option change number of processors for mapping (1-8)

Edit file : tophat3.pl (for Poplar version3)

```
use strict;

# input files

my @fastqfiles = glob ("/media/Archive1/fastq/tissue_PO/*.fastq");
my $seqindex = "/genomes/ptrichocarpa/index/Ptrichocarpa_210.fa";
my $GFF = "/genomes/ptrichocarpa/sequences/Ptrichocarpa_210_gene_exons.gff3";

# output directory

my $outdir = "/media/Archive1/BAM/v3.0/tissue_PO/";
mkdir $outdir;

# Process each file

foreach my $qfile (@fastqfiles){
    my @aa = split /\//,$qfile;

    my $filename = pop @aa;
    my @bb = split /\./,$filename;
    my $addir = shift @bb;
    my $outdir=$outdir.$addir;

    print "\nprocesseing $qfile\n";
    print "tophat -p 7 -G $GFF -o $outdir $seqindex $qfile\n";

}

# Using 6 processors

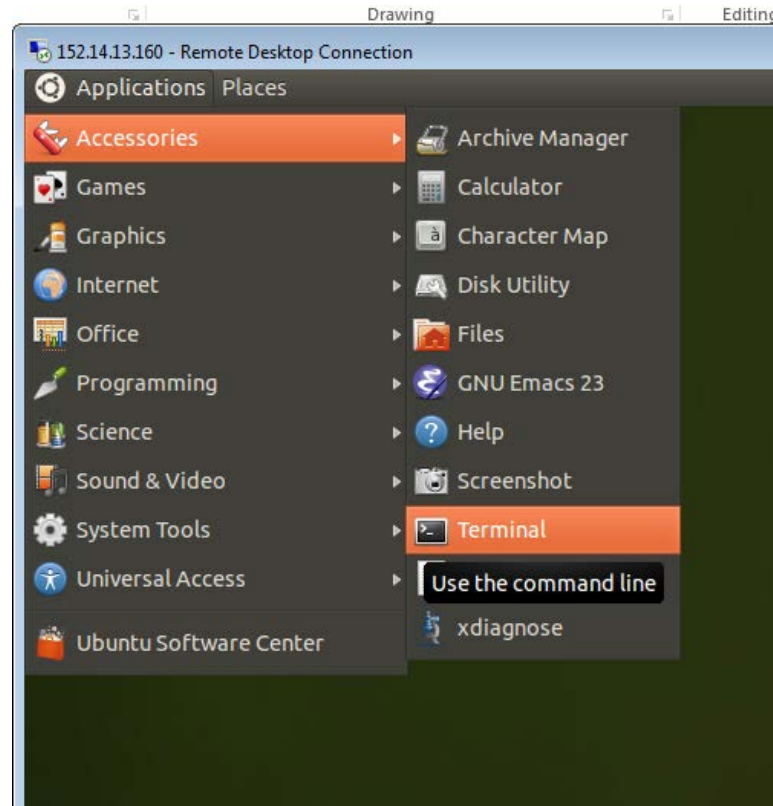
system "tophat -p 7 -G $GFF -o $outdir $seqindex $qfile";
```

Change the
location of fastq
files

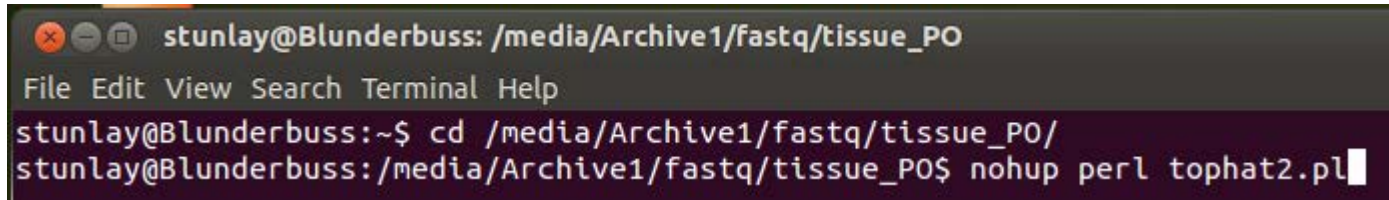
Change the
location of
result BAM
files

Option change number of
processors for mapping (1-8)

Execute using terminal in Ubuntu Linux



Execute using terminal in Ubuntu Linux

A screenshot of a terminal window. The title bar shows the user 'stunlay@Blunderbuss' and the current directory '/media/Archive1/fastq/tissue_PO'. The terminal content shows the user navigating to the directory and running a Perl script. The prompt is 'stunlay@Blunderbuss:~\$' and the command is 'cd /media/Archive1/fastq/tissue_PO/'. The next prompt is 'stunlay@Blunderbuss:/media/Archive1/fastq/tissue_PO\$' and the command is 'nohup perl tophat2.pl'.

```
stunlay@Blunderbuss: /media/Archive1/fastq/tissue_PO
File Edit View Search Terminal Help
stunlay@Blunderbuss:~$ cd /media/Archive1/fastq/tissue_PO/
stunlay@Blunderbuss:/media/Archive1/fastq/tissue_PO$ nohup perl tophat2.pl
```

go to directory of file tophat2.pl

```
cd /media/Archive1/fastq/tissue_PO/
```

#nohup used to store the screen output into nohup.out

#perl used to execute the perl program for run file tophat.pl

```
nohup perl tophat2.pl
```


Example of log file in step 1

```
processeing /media/Archive1/fastq/tissue_PO/GSL225_01-VC_PtX1_NCSU_CGATGT_L001_R1_001.fastq
tophat -p 7 -G /genomes/ptrichocarpa/sequencesV2.2/Ptrichocarpa_156_gene_exons.gff3 -o /media/Archive1/BAM/v2.2/tissue_PO/GSL225_01-VC_PtX1_NCSU_CGATGT_L001_R1_001
/genomes/ptrichocarpa/indexV2.2/bowtie2-index/Ptrichocarpa_156.fa /media/Archive1/fastq/tissue_PO/GSL225_01-VC_PtX1_NCSU_CGATGT_L001_R1_001.fastq
```

```
[2014-03-21 17:02:23] Beginning TopHat run (v2.0.3)
```

```
[2014-03-21 17:02:23] Checking for Bowtie
```

```
      Bowtie version:      2.0.0.6
```

```
[2014-03-21 17:02:23] Checking for Samtools
```

```
      Samtools version:    0.1.18.0
```

```
[2014-03-21 17:02:24] Checking for Bowtie index files
```

```
[2014-03-21 17:02:24] Checking for reference FASTA file
```

```
      Warning: Could not find FASTA file /genomes/ptrichocarpa/indexV2.2/bowtie2-index/Ptrichocarpa_156.fa
```

```
[2014-03-21 17:02:24] Reconstituting reference FASTA file from Bowtie index
```

```
      Executing: /usr/local/bin/bowtie2-inspect /genomes/ptrichocarpa/indexV2.2/bowtie2-index/Ptrichocarpa_156.fa > /media/Archive1/BAM/v2.2/tissue_PO/GSL225_01-
VC_PtX1_NCSU_CGATGT_L001_R1_001/tmp/Ptrichocarpa_156.fa
```

```
[2014-03-21 17:02:47] Generating SAM header for /genomes/ptrichocarpa/indexV2.2/bowtie2-index/Ptrichocarpa_156.fa
```

```
      format:                phred33 (default)      fastq
```

```
      quality scale:         phred33 (default)
```

```
[2014-03-21 17:02:49] Reading known junctions from GTF file
```

```
[2014-03-21 17:02:52] Preparing reads
```

```
      left reads: min. length=100, max. length=100, 20010819 kept reads (255 discarded)
```

```
[2014-03-21 17:10:26] Creating transcriptome data files..
```

```
[2014-03-21 17:10:33] Building Bowtie index from Ptrichocarpa_156_gene_exons.fa
```

```
[2014-03-21 17:14:05] Mapping left_kept_reads to transcriptome Ptrichocarpa_156_gene_exons with Bowtie2
```

```
[2014-03-21 17:27:34] Converting left_kept_reads.m2g to genomic coordinates (map2gtf)
```

```
[2014-03-21 17:36:21] Resuming TopHat pipeline with unmapped reads
```

```
[2014-03-21 17:36:57] Mapping left_kept_reads.m2g_um to genome Ptrichocarpa_156.fa with Bowtie2
```

```
[2014-03-21 17:44:18] Mapping left_kept_reads.m2g_um_seg1 to genome Ptrichocarpa_156.fa with Bowtie2 (1/4)
```

```
[2014-03-21 17:46:39] Mapping left_kept_reads.m2g_um_seg2 to genome Ptrichocarpa_156.fa with Bowtie2 (2/4)
```

```
[2014-03-21 17:49:04] Mapping left_kept_reads.m2g_um_seg3 to genome Ptrichocarpa_156.fa with Bowtie2 (3/4)
```

```
[2014-03-21 17:51:29] Mapping left_kept_reads.m2g_um_seg4 to genome Ptrichocarpa_156.fa with Bowtie2 (4/4)
```

```
[2014-03-21 17:53:39] Searching for junctions via segment mapping
```

```
[2014-03-21 17:55:55] Retrieving sequences for splices
```

```
[2014-03-21 17:56:18] Indexing splices
```

```
[2014-03-21 17:57:26] Mapping left_kept_reads.m2g_um_seg1 to genome segment_juncs with Bowtie2 (1/4)
```

```
[2014-03-21 17:58:24] Mapping left_kept_reads.m2g_um_seg2 to genome segment_juncs with Bowtie2 (2/4)
```

```
[2014-03-21 17:59:23] Mapping left_kept_reads.m2g_um_seg3 to genome segment_juncs with Bowtie2 (3/4)
```

```
[2014-03-21 18:00:24] Mapping left_kept_reads.m2g_um_seg4 to genome segment_juncs with Bowtie2 (4/4)
```

```
[2014-03-21 18:01:19] Joining segment hits
```

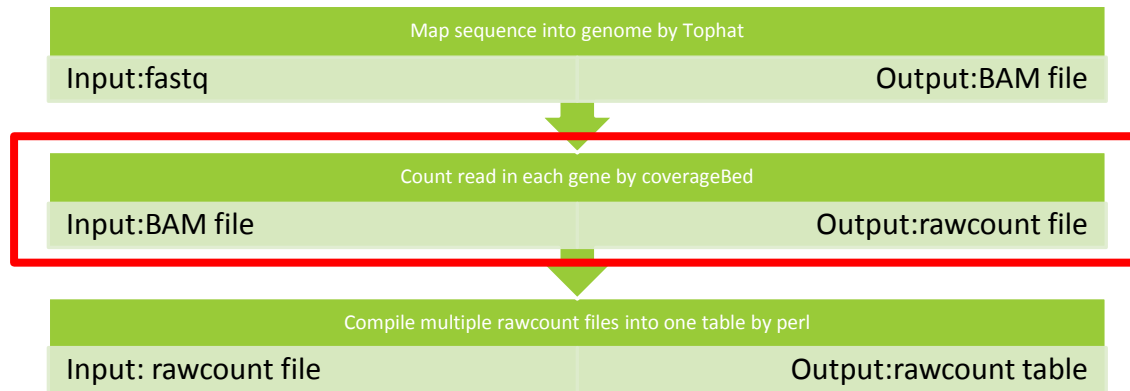
```
[2014-03-21 18:17:52] Reporting output tracks
```

```
[2014-03-21 18:29:24] Run complete: 01:27:01 elapsed
```

What is the result (BAM) file look like?

```
/media/Archive1/BAM/v2.2/tissue_PO/GSL226_11-VC_k19_NCSU_GTCCGC_L002_R1_001/accepted_hits.bam - u
Encoding
< ␣ ỹ- BC₁ 0} ÈŠ6AR†u9^_ - •4yø•š
†: "lne<€ ȳ ÷ Y 0¶Āy|½êEtōw±†|ūē!BøEzø0?b³?yã?úózb«BūAī~[bīùfiyî·yü‡?ūxžúū: b
ž44æøB ? çÑĐE- :ê[ižúòš%ŌçŌ† ‡žī-†ç^††Q|Yx•ao¶%øP%Ō1fXúH`Am!&B}€f†G-÷
†¶pÆHŪ¶w¶ f⊙.=
IŌ+· ¶ E¶ 3Ā3†x@J@ E =v4ª†=ú-
9
è¹¥) ðŪ>=-èy~ë,•=oqb_¥ † s7| ð\iĀJA -¶ Ĩ>†¶ †»z9 ù}0{-f wū|÷EEÚD ²Đ¶ €%...
¶ -† 7nŌ• :J(ætÉĀACŸM9† x/.iB¥ KYaB¥âum_*èàŌ†°ŌY
† †. €àGŌ•p~
‡|@ðfē††ūè†|†?@„cDYM}_i³·|u(èàb[09 x#vŪ ††-†,•çŌ¥_-æ0
<•fs»†|:
8wZ~ æ0-:è~»° Ašâ;9 ðpà\
ŪĒĀ¹}5²%Nð«- Ĩ-Ÿ>†P¶Y Ū¶"Ÿ1µp_¶†X(àŪUM80¶5°fsŪz†" æ»f!{Ū †|PŌ-²G·iłsŪtaĀšêm
òšKŪàŪ>uĒš† qšæ¶ ††-†%štk™GEĀ,βZŪ†t%íuf Ĩ"i"ªCŪY÷f
óhhè_
@n_šYÆøP"-ª)• ÷NŪxŌ=*
%šÉ†òŪ,l. • w‡=• o;^y"ŌŌK |üĨ&\X,ūæx<†¶ «Ū-†‡:*ð«qðq †š-†çìà9ĒĀøqæš]´! v|{†
ŪĨ«-†T. ĒĀ»%.:SĀĀçY®,7ø4ĀNP·Ÿ#ĀZŸŌ° ,
D† ð\
†ĀŪx²E†ççöf-D oó>àè,...Ō,†- ¶ j9Ÿ8È°èæ- h%ŌàžŪ%yx{c"z†- ^Æ Y%>ŸšL»·PŸ^%j9"8"»x¶;
Ūø²BŌŪ*; † %8 %šŪfe† Ō-u,‰%ž AZ_´ĀĀY=Ō;@ŸYUX" āb Ō=>¥èā ĨžŪ±AYŌ;KĀĒ- |<+»:
écĀéŌ.€fx»βŌàðwŌ†x=,B
ðZ†i,xĀ v %pI´Ū
jĀĀé]á] ~,•- ò° ?Ā|æ†Ō-
÷†kx3€ēē÷žyy
~
xu †pwèlBšUúĒ¶\iE†‡xªæed÷†éŌ ,_,|= [āĒ VĨª´|èù<x%|ĀŌ`Ū Ū>GC :F´ūĒ†‡÷ ,ŸX.
†4F=H;†m-Ē#%@žŪŪ†
Ō
¶%xp± "†P a;¶ Ū†@†lŌŪcñC₁ ð<aæ† èk:,ofD...c5xL{ €pçŪŌ è?¶Ō%@žŌV- ŸC,‰$KĀ,M"ā, -½
`Ūn`Ō ,βŪŪ āŪ`w Ē7}†fāg±|xvŌPŌ† Ūmzn*Ō†ŸāŌ-hŸYq† IT}ŪŌfðēyē-†,†C}¶ŌçĒ%·iYç$
```

View in IGV



Step2

CONVERT BAM FILE TO RAWCOUNT FILE

File : get_raw_v2.pl

```
use strict;

# input files

my $gff = "/genomes/trichocarpa/sequencesV2.2/Trichocarpa_156_gene_exons.gff3";
my @BAMfile = glob ("/media/Archive1/BAM/v2.2/tissue_PO/*/accepted_hits.bam");

# output file
my $outdir = "/media/Archive1/coverageBed/v2.2/tissue_PO/";

mkdir $outdir;

foreach my $bfile (@BAMfile){


    #if ($pac eq "tophat"){

    print "$bfile\n";
    my @aa = split /\//,$bfile;
    pop @aa;
    my $tfile = pop @aa;


    print "coverageBed -abam $bfile -b $gff > $outdir/$tfile.rawcount\n";
    system "coverageBed -abam $bfile -b $gff > $outdir/$tfile.rawcount";

    }
}
```

Change the
location of
result BAM
files



Change the location of output
rawcount files



File : get_raw_v3.pl

```
use strict;

# input files

my $gff = "/genomes/ptrichocarpa/sequences/Ptrichocarpa_210_gene_exons.gff3";
my @BAMfile = glob ("/media/Archive1/BAM/v3.0/tissue_PO/*/accepted_hits.bam");

# output file
my $outdir = "/media/Archive1/coverageBed/v3.0/tissue_PO/";

mkdir $outdir;

foreach my $bfile (@BAMfile){


    #if ($pac eq "tophat"){ $bfile = $bfile . "/accepted_hits.bam"}

    print "$bfile\n";
    my @aa = split /\//, $bfile;
    pop @aa;
    my $tfile = pop @aa;


    print "coverageBed -abam $bfile -b $gff > $outdir/$tfile\.rawcount\n";
    system "coverageBed -abam $bfile -b $gff > $outdir/$tfile\.rawcount";

}
```

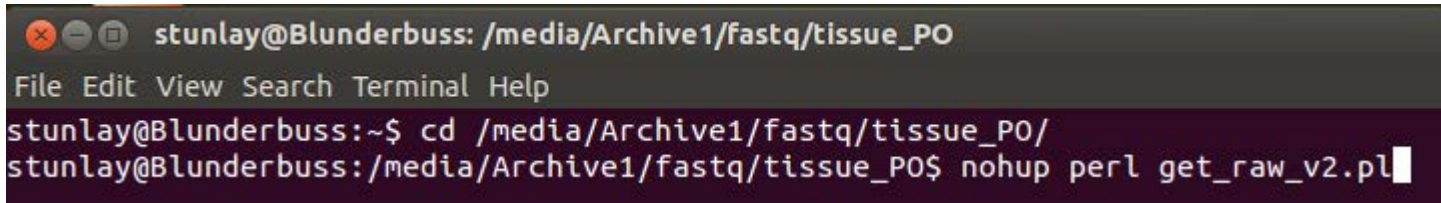
Change the
location of
result BAM
files



Change the location of output
rawcount files



Execute using terminal in Ubuntu Linux



```
stunlay@Blunderbuss: /media/Archive1/fastq/tissue_PO
File Edit View Search Terminal Help
stunlay@Blunderbuss:~$ cd /media/Archive1/fastq/tissue_PO/
stunlay@Blunderbuss:/media/Archive1/fastq/tissue_PO$ nohup perl get_raw_v2.pl
```

#nohup used to store the screen output into nohup.out

#perl used to execute the perl program for run file get_raw_v2.pl

nohup perl get_raw_v2.pl

Example of log file

```
/media/Archive1/BAM/v2.2/tissue_PO/GSL225_01-  
VC_PtX1_NCSU_CGATGT_L001_R1_001/accepted_hits.bam
```

```
coverageBed -abam /media/Archive1/BAM/v2.2/tissue_PO/GSL225_01-  
VC_PtX1_NCSU_CGATGT_L001_R1_001/accepted_hits.bam -b  
/genomes/ptrichocarpa/sequencesV2.2/Ptrichocarpa_156_gene_exons.gff3 >  
/media/Archive1/coverageBed/v2.2/tissue_PO//GSL225_01-  
VC_PtX1_NCSU_CGATGT_L001_R1_001.rawcount
```

```
/media/Archive1/BAM/v2.2/tissue_PO/GSL225_02-  
VC_PtX2_NCSU_TGACCA_L001_R1_001/accepted_hits.bam
```

```
coverageBed -abam /media/Archive1/BAM/v2.2/tissue_PO/GSL225_02-  
VC_PtX2_NCSU_TGACCA_L001_R1_001/accepted_hits.bam -b  
/genomes/ptrichocarpa/sequencesV2.2/Ptrichocarpa_156_gene_exons.gff3 >  
/media/Archive1/coverageBed/v2.2/tissue_PO//GSL225_02-  
VC_PtX2_NCSU_TGACCA_L001_R1_001.rawcount
```

How coverageBed work?

coverageBed computes both the depth and breadth of coverage of features in **file A** across the features in **file B**

Usage: \$ coverageBed [OPTIONS] -a <BED> -b <BED>

In our case we specify **BAM file as A** and **GFF file as B**, so we will count depth of each feature in GFF file



Output file : *.rawcount

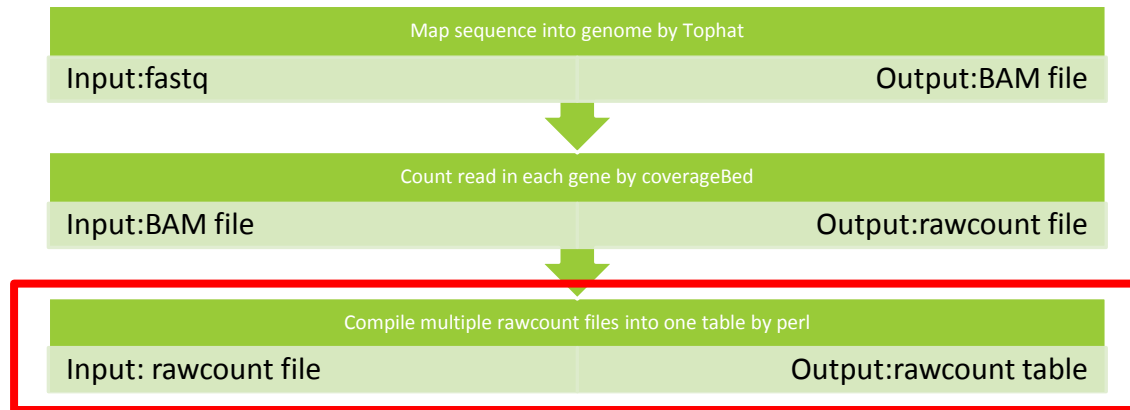
A	B	C	D	E	F	G	H	I	J	K	L	M
scaffold_1	phytozome8_0	gene	8387424	8394306	.	+	.	ID=POPTR_0001s10780;Name=POPTR_0001s10780	267	6661	6883	0.9677466
scaffold_1	phytozome8_0	mRNA	8387424	8394306	.	+	.	ID=PAC:18234938;Name=POPTR_0001s10780.1;pacid	267	6661	6883	0.9677466
scaffold_1	phytozome8_0	gene	4189325	4195866	.	-	.	ID=POPTR_0001s05460;Name=POPTR_0001s05460	11	3665	6542	0.5602262
scaffold_1	phytozome8_0	mRNA	4189325	4195866	.	-	.	ID=PAC:18237393;Name=POPTR_0001s05460.1;pacid	11	3665	6542	0.5602262
scaffold_1	phytozome8_0	gene	14676188	14680501	.	-	.	ID=POPTR_0001s17800;Name=POPTR_0001s17800	128	1985	4314	0.4601298
scaffold_1	phytozome8_0	mRNA	14676188	14680501	.	-	.	ID=PAC:18234264;Name=POPTR_0001s17800.1;pacid	128	1985	4314	0.4601298
scaffold_1	phytozome8_0	gene	20970609	20973083	.	-	.	ID=POPTR_0001s22350;Name=POPTR_0001s22350	183	2272	2475	0.9179798

J = number of read count in feature (raw count)

K = The number of bases coverage from BAM file in features of GFF file

L = The length of the feature in B.

M = K/L



Step 3

COMPILE THE MULTIPLE RAWCOUNT FILES INTO ONE TABLE BY PERL

File : compile_mRNA_v2.pl

```
my @expressionfile = glob ("/media/Archive1/coverageBed/v2.2/tissue_PO/*.rawcount");

foreach my $exfiles (@expressionfile){
    my @bb = split /\//,$exfiles;

    my $tline = pop @bb;
    $tline =~ s/\.rawcount//g;
    $samples{$tline}=1;
    print "$exfiles\t$tline\n";

# open each input .rawcount file

    open IN,"$exfiles"||die;
    while (<IN>){
        chomp;
        my $line = $_;
        my @aa = split /\t/, $line;
        my @cc = split /;/,$aa[8];
        my $gene = $cc[1];
        $gene =~ s/Name=//g;
        if ($aa[2] eq "mRNA"){
            $H{$gene,$tline}=$aa[9];
            #print "$gene\t$tline\t$aa[9]\n"
        }
    }

# output file
    open out,">/media/Archive2/rawcount/mRNA_v2_tissue_PO.rawtable"||die;
```

Change the location of rawcount files

Option change to gene or
exon for different analysis

Change the location and name of output rawcount table files

File : compile_mRNA_v3.pl

```
my @expressionfile = glob ("/media/Archive1/coverageBed/v3.0/tissue_PO/*.rawcount");

foreach my $exfiles (@expressionfile){
    my @bb = split /\//,$exfiles;

    my $tline = pop @bb;
    $tline =~ s/\.rawcount//g;
    $samples{$tline}=1;
    print "$exfiles\t$tline\n";

# open each input .rawcount file

    open IN,"$exfiles"||die;
    while (<IN>){
        chomp;
        my $line = $_;
        my @aa = split /\t/, $line;
        my @cc = split /;/,$aa[8];
        my $gene = $cc[1];
        $gene =~ s/Name//g;
        if ($aa[2] eq "mRNA"){
            $H{$gene,$tline}=$aa[9];
            #print "$gene\t$tline\t$aa[9]\n"
        }
    }

# output file

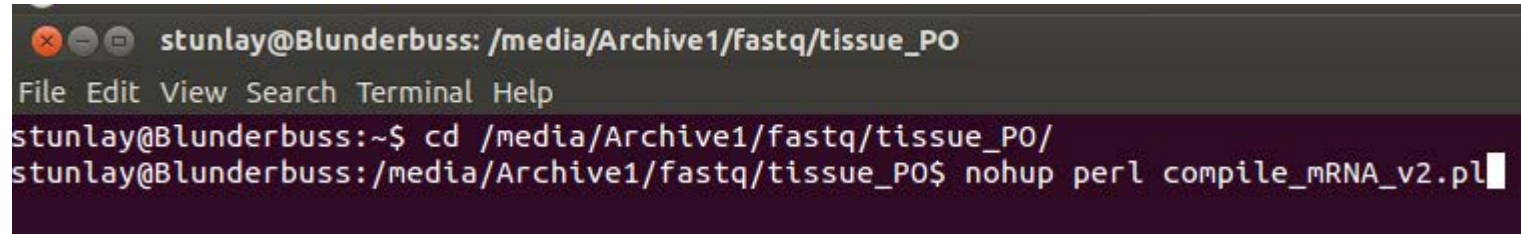
    open out,">/media/Archive2/rawcount/mRNA_v3_tissuePO.rawtable"||die;
```

Change the location of rawcount files

Option change to gene or
exon for different analysis

Change the location and name of output rawcount table files

Execute using terminal in Ubuntu Linux

A terminal window screenshot with a dark background. The title bar shows 'stunlay@Blunderbuss: /media/Archive1/fastq/tissue_PO'. The terminal content shows the user navigating to the directory and running a command. The prompt changes from '~\$' to '/media/Archive1/fastq/tissue_PO\$' after the 'cd' command. The command 'nohup perl compile_mRNA_v2.pl' is entered and executed, with a cursor at the end of the line.

```
stunlay@Blunderbuss: /media/Archive1/fastq/tissue_PO
File Edit View Search Terminal Help
stunlay@Blunderbuss:~$ cd /media/Archive1/fastq/tissue_PO/
stunlay@Blunderbuss:/media/Archive1/fastq/tissue_PO$ nohup perl compile_mRNA_v2.pl
```

#nohup used to store the screen output into nohup.out

#perl used to execute the perl program for run file
compile_mRNA_v2.pl

nohup perl compile_mRNA_v2.pl

Output file : *.rawtable

gene	GSL226_09-VC_k8_NCSU_AGTTC_L002_R1_001	GSL226_10-VC_k9_NCSU_ATGTCA_L002_R1_001	GSL226_11-VC_k19_NCSU_GTCCGC_L002_R1_001
POPTR_0001s10780.1	347	391	267
POPTR_0001s10790.1	137	118	84
POPTR_0001s10800.1	155	186	164
POPTR_0001s10810.1	18	27	9
POPTR_0001s10810.2	18	27	9
POPTR_0001s10810.3	18	27	9
POPTR_0001s10810.4	18	27	9

Step4

IDENTIFY DIFFERENTIAL EXPRESSED GENES BY
EDGER

File : edgeR_Pop_tissue_BGI_workshop.R

```
> setwd("~/RNASeq_workshop")
> #command to install package edgeR
> source("http://bioconductor.org/biocLite.R")
Bioconductor version 2.12 (BiocInstaller 1.10.3), ?biocLite for help
> biocLite("edgeR")
trying URL 'http://bioconductor.org/packages/2.12/bioc/bin/windows/contrib/3.0/BiocInstaller_1.10.4.zip'
Content type 'application/zip' length 50486 bytes (49 kb)
opened URL
downloaded 49 kb
```

The downloaded binary packages are in

```
C:\Users\stunlay\AppData\Local\Temp\RtmpcFGp0r\downloaded_packages
Bioconductor version 2.12 (BiocInstaller 1.10.4), ?biocLite for help
A newer version of Bioconductor is available for this version of R, ?BiocUpgrade for help
'BiocInstaller' updated to version 1.10.4
Bioc_mirror: http://bioconductor.org
Using Bioconductor version 2.12 (BiocInstaller 1.10.4), R version 3.0.2.
Installing package(s) 'edgeR'
warning: package 'edgeR' is in use and will not be installed
Old packages: 'arm', 'BiasedUrn', 'bit', 'caTools', 'colorspace', 'devtools', 'digest', 'doParallel', 'dynamicTreeCut',
  'e1071', 'edgeR', 'evaluate', 'ff', 'GGally', 'ggbio', 'gplots', 'gstat', 'gtools', 'Hmisc', 'httr', 'hydroGOF',
  'hydrotSM', 'igraph', 'iplots', 'lavaan', 'limma', 'lme4', 'lsmeans', 'mapproj', 'minqa', 'mixomics', 'mnormt',
  'multcomp', 'mvtnorm', 'nlS2', 'pastecs', 'plotrix', 'plyr', 'png', 'Rcpp', 'RcppArmadillo', 'RcppEigen', 'rgl', 'Rgtk2',
  'rJava', 'segmented', 'seriation', 'sp', 'VariantAnnotation', 'WGCNA', 'xtable', 'zoo'
update all/some/none? [a/s/n]:
n
warning message:
installed directory not writable, cannot update packages 'boot', 'cluster', 'foreign', 'KernSmooth', 'lattice', 'MASS',
  'Matrix', 'mgcv', 'nlme', 'rpart', 'survival'
> library(edgeR)
> #import file "BGI_tissue.txt" (rawcount table) into dataframe "y"
> #header= T => table has header
> #sep="\t" => specify as tab delimited or sepeerate column by tab
> #row.names=1 => first column is row names
> y <- read.table("BGI_tissue.txt", header=T, sep="\t", row.names=1)
> head(y)
```

	Pt_X1	Pt_X2	Pt_X3	Pt_P1	Pt_P2	Pt_P3	Pt_L1	Pt_L2	Pt_L3	Pt_S1	Pt_S2	Pt_S3
POPTR_0001s00200.1	7	9	7	12	6	2	9	3	5	13	4	3
POPTR_0001s00210.1	155	131	94	96	82	44	42	93	92	130	67	89
POPTR_0001s00220.1	104	57	46	4	15	0	19	12	20	13	12	13
POPTR_0001s00230.1	0	0	1	0	0	0	0	0	0	0	0	0
POPTR_0001s00240.1	39	0	6	58	1	12	1	2	7	192	45	28
POPTR_0001s00250.1	7404	4088	5891	3312	3361	1317	417	689	994	2343	1145	1519


```

> #specify the group identifier into object "group"
> #rep(1:4,each=3) = 1,1,1,2,2,2,3,3,3,4,4,4
> group <- rep(1:4,each=3)
> group
[1] 1 1 1 2 2 2 3 3 3 4 4 4
> #put rawcount and group into the edgeR format
> tis <- DGEList(count = y, group = group)
> # Calculate Normalization Factor using TMM by Robinson MD,
> # Oshlack A (2010). Genome Biology 11, R25.
> tis <- calcNormFactors(tis,method="TMM")
> tis
An object of class "DGEList"
$counts
      Pt_X1 Pt_X2 Pt_X3 Pt_P1 Pt_P2 Pt_P3 Pt_L1 Pt_L2 Pt_L3 Pt_S1 Pt_S2 Pt_S3
POPTR_0001s00200.1      7      9      7     12      6      2      9      3      5     13      4      3
POPTR_0001s00210.1    155    131     94     96     82     44     42     93     92    130     67     89
POPTR_0001s00220.1    104     57     46      4     15      0     19     12     20     13     12     13
POPTR_0001s00230.1      0      0      1      0      0      0      0      0      0      0      0      0
POPTR_0001s00240.1     39      0      6     58      1     12      1      2      7    192     45     28
45028 more rows ...

$samples
      group lib.size norm.factors
Pt_X1     1 23215783   1.030617
Pt_X2     1 16455445   1.071571
Pt_X3     1 14644964   1.100784
Pt_P1     2 14200511   1.259262
Pt_P2     2 14935270   1.251318
7 more rows ...

> tis$samples
      group lib.size norm.factors
Pt_X1     1 23215783   1.0306175
Pt_X2     1 16455445   1.0715715
Pt_X3     1 14644964   1.1007844
Pt_P1     2 14200511   1.2592618
Pt_P2     2 14935270   1.2513180
Pt_P3     2 12363172   0.6896963
Pt_L1     3 16830260   0.7132996
Pt_L2     3 19568204   0.7951466
Pt_L3     3 20417321   0.8153829
Pt_S1     4 20321791   1.2454369
Pt_S2     4 10827664   1.1617584
Pt_S3     4 12109146   1.1311473
> |

```

```

> #calculate normalize count using function cpm (count per million)
> nc.tis <- cpm(tis, normalized.lib.sizes=TRUE)
> #write the normalize count into file "nc_tissue.csv"
> write.csv(nc.tis, file = "nc_tissue.csv")
> head(nc.tis)

```

	Pt_X1	Pt_X2	Pt_X3	Pt_P1	Pt_P2	Pt_P3	Pt_L1	Pt_L2
POPTR_0001s00200.1	0.2925615	0.5104013	0.43421764	0.6710598	0.32104839	0.2345536	0.7496865	0.1928071
POPTR_0001s00210.1	6.4781480	7.4291739	5.83092263	5.3684786	4.38766133	5.1601801	3.4985369	5.9770212
POPTR_0001s00220.1	4.3466283	3.2325413	2.85343022	0.2236866	0.80262098	0.0000000	1.5826715	0.7712285
POPTR_0001s00230.1	0.0000000	0.0000000	0.06203109	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
POPTR_0001s00240.1	1.6299856	0.0000000	0.37218655	3.2434558	0.05350807	1.4073218	0.0832985	0.1285381
POPTR_0001s00250.1	309.4465019	231.8355950	365.42516161	185.2125127	179.84060657	154.4535715	34.7354738	44.2813718

```


```

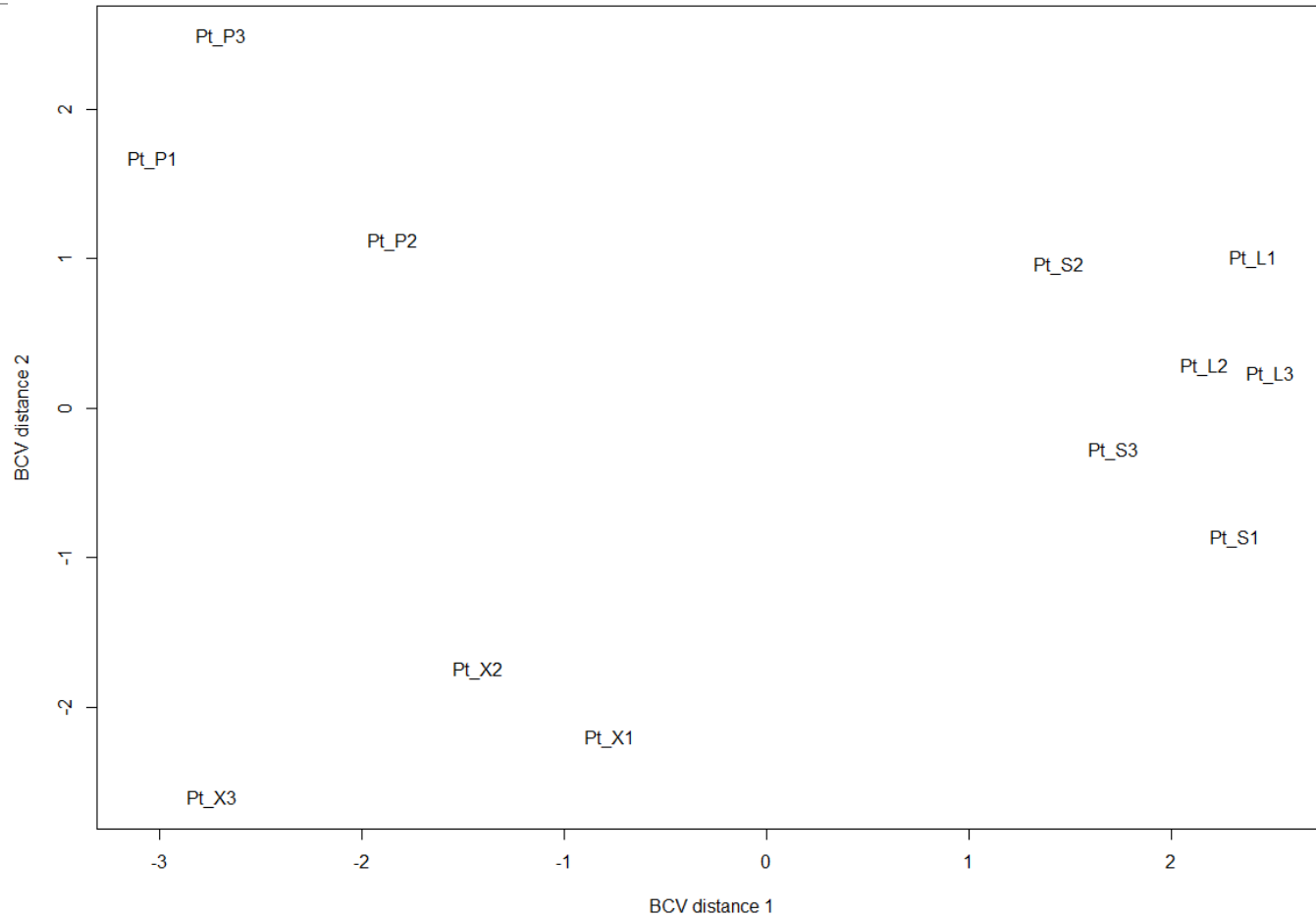
	Pt_L3	Pt_S1	Pt_S2	Pt_S3
POPTR_0001s00200.1	0.3003376	0.5136409	0.3179870	0.2190224
POPTR_0001s00210.1	5.5262112	5.1364095	5.3262827	6.4976654
POPTR_0001s00220.1	1.2013503	0.5136409	0.9539611	0.9490972
POPTR_0001s00230.1	0.0000000	0.0000000	0.0000000	0.0000000
POPTR_0001s00240.1	0.4204726	7.5860817	3.5773541	2.0442094
POPTR_0001s00250.1	59.7071077	92.5739032	91.0237864	110.8983573

```

> #estimate the common dispersion to get overall degree of inter-library variability in the data
> tis <- estimateCommonDisp(tis,verbose=TRUE)
Disp = 0.08324 , BCV = 0.2885
> #estimate the tagwise dispersion
> tis <- estimateTagwiseDisp(tis)
> #check the relationship between sample by plot multi dimension scaling
> plotMDS(tis, method="bcv")
>

```

Plot MDS show 3 clusters of different tissues



```

> d_lf_xy <- exactTest(tis,pair=c("3","1"))
> d_lf_xy
An object of class "DGEEexact"
$table
      logFC      logCPM      PValue
POPTR_0001s00200.1 0.03230106 -0.9374979 1.000000000
POPTR_0001s00210.1 0.38686578  2.5076883 0.186018361
POPTR_0001s00220.1 1.56321759  0.7109259 0.002781238
POPTR_0001s00230.1 1.77311261 -2.9891833 1.000000000
POPTR_0001s00240.1 1.61899419  0.9354239 0.181550504
45028 more rows ...

$comparison
[1] "3" "1"

$genes
NULL

> toptag_lf_xy <- topTags(d_lf_xy, n=Inf)
> head(toptag_lf_xy$table)
      logFC      logCPM      PValue      FDR
POPTR_0007s13720.1  8.505188  8.548637 4.839798e-112 2.179506e-107
POPTR_0006s06010.1 -9.919535  3.994338 5.785553e-101 1.302704e-96
POPTR_0016s14310.1 -7.672346  7.401097 6.341108e-100 9.518637e-96
POPTR_0001s26630.1 -8.758553  5.693328 1.887435e-97  2.124921e-93
POPTR_0001s02770.1 -8.131885  6.120178 4.592215e-97  4.136024e-93
POPTR_0016s05780.1 -8.702961  5.658648 1.753710e-95  1.316247e-91
> |
> toptag_lf_xy <- toptag_lf_xy$table[order(rownames(toptag_lf_xy$table)),]
> head(toptag_lf_xy)
      logFC      logCPM      PValue      FDR
POPTR_0001s00200.1 0.03230106 -0.9374979 1.000000e+00 1.000000e+00
POPTR_0001s00210.1 0.38686578  2.5076883 1.860184e-01 3.040310e-01
POPTR_0001s00220.1 1.56321759  0.7109259 2.781238e-03 6.946147e-03
POPTR_0001s00230.1 1.77311261 -2.9891833 1.000000e+00 1.000000e+00
POPTR_0001s00240.1 1.61899419  0.9354239 1.815505e-01 2.978529e-01
POPTR_0001s00250.1 2.70719027  7.2774809 1.332564e-18 1.347012e-17
> write.table(toptag_lf_xy, file = "FC_lf_xy.txt", sep = "\t", col.names=NA)
> |

```

Interpret the result

	logFC	logCPM	PValue	FDR
POPTR_0001s00200.1	0.03	-0.94	1.00	1.00
POPTR_0001s00210.1	0.39	2.51	0.19	0.30
POPTR_0001s00220.1	1.56	0.71	0.00	0.01
POPTR_0001s00230.1	1.77	-2.99	1.00	1.00
POPTR_0001s00240.1	1.62	0.94	0.18	0.30

LogFC = log base 2 of expression xylem compare to leaf
(+ expression in xylem more than leaf ,
- expression in xylem less than leaf)

LogCPM = log base 2 of average expression from all samples

Pvalue = P value associate with Log FC (ignore because multiple testing)

FDR = corrected P value for multiple testing

Using Pop's pipes sys.bio.mtu.edu/deg.php



← → ↻ sys.bio.mtu.edu/deg.php

Apps ๒๒ มีนาคม ๒๕๖๑! ๒๒:๑๓... + Sermsawat Yumeiro Patissiere e... NCDOT : ทบงกรมการขนส่งทางบก นำเข้าจาก Safari Khan Academy » Other book

Pop's pipes: Poplar Gene Expression Data Analysis Pipelines

Home Identification of DEGs Pathway Enrichment Domain Enrichment GO Enrichment GO-Tree Other Softwares Manual Register Logout
Sample Input and Output files

Welcome stunlay

To logout [Click here!](#)

Identification of Differentially Expressed Genes(DEGs)

1. Select data platform type:

RNA-seq ▾

1.1 Select RNA-seq normalization Method (RNA-seq only)

edgeR ▾

Input file : tab delimited and specify control as C and treatment as T

gene	Pt_X1	Pt_X2	Pt_X3	Pt_P1	Pt_P2	Pt_P3	Pt_X1	Pt_X2	Pt_X3	Pt_L1	Pt_L2	Pt_L3	Pt_X1	Pt_X2	Pt_X3	Pt_S1	Pt_S2	Pt_S3
gene	C	C	C	T1	T1	T1	C	C	C	T2	T2	T2	C	C	C	T3	T3	T3
Potri.001G000100.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Potri.001G000200.1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Potri.001G000300.1	301	181	240	246	133	98	301	181	240	66	59	94	301	181	240	142	153	183
Potri.001G000400.1	451	258	309	293	184	117	451	258	309	99	85	118	451	258	309	222	214	278
Potri.001G000400.2	725	482	516	469	296	205	725	482	516	152	140	205	725	482	516	371	332	546
Potri.001G000400.3	644	425	447	425	268	169	644	425	447	127	124	165	644	425	447	319	285	484
Potri.001G000400.4	643	425	446	423	268	169	643	425	446	126	124	164	643	425	446	318	285	481
Potri.001G000500.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	1	0
Potri.001G000600.1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	3	0	0
Potri.001G000700.1	1246	888	839	472	642	225	1246	888	839	358	495	590	1246	888	839	962	593	1136
Potri.001G000700.2	1246	888	839	472	642	225	1246	888	839	358	495	590	1246	888	839	962	593	1136
Potri.001G000800.1	383	183	144	4	5	0	383	183	144	25	49	43	383	183	144	384	74	446
Potri.001G000900.1	3664	2513	2578	2457	2130	1184	3664	2513	2578	1397	1400	1908	3664	2513	2578	2142	1898	1941
Potri.001G000900.2	3781	2588	2651	2515	2208	1227	3781	2588	2651	1460	1465	1997	3781	2588	2651	2257	2000	2037
Potri.001G000900.3	3781	2588	2651	2515	2208	1227	3781	2588	2651	1460	1465	1997	3781	2588	2651	2257	2000	2037
Potri.001G000900.4	3781	2588	2651	2515	2208	1227	3781	2588	2651	1460	1465	1997	3781	2588	2651	2257	2000	2037
Potri.001G000900.5	3781	2588	2651	2515	2208	1227	3781	2588	2651	1460	1465	1997	3781	2588	2651	2257	2000	2037
Potri.001G000900.6	3781	2588	2651	2515	2208	1227	3781	2588	2651	1460	1465	1997	3781	2588	2651	2257	2000	2037
Potri.001G001000.1	0	0	0	0	0	0	0	0	0	22	32	63	0	0	0	1	4	0

Original label not include in the input file

1 group of testing

Example of output file

GeneID	Treat_1_FC	Treat_1_pvalue	Treat_1_corrected pvalue(FDR)	DEG/nonDEG	Treat_2_FC	Treat_2_pval	Treat_2_corr
Potri.001G000100.1	0	1	1	nonDEG	0	1	1
Potri.001G000200.1	2.052966579	1	1	nonDEG	2.053572251	1	1
Potri.001G000300.1	-0.108360834	0.721937035	0.905099598	nonDEG	-1.24673887	4.43E-05	0.000131252
Potri.001G000400.1	-0.283314192	0.284932071	0.446710298	nonDEG	-1.25342151	8.28E-06	2.67E-05
Potri.001G000400.2	-0.324675777	0.208482818	0.348368372	nonDEG	-1.31037424	3.54E-07	1.33E-06
Potri.001G000400.3	-0.326804745	0.197965959	0.333358439	nonDEG	-1.37923567	6.98E-08	2.82E-07
Potri.001G000400.4	-0.327745289	0.195890129	0.330350526	nonDEG	-1.38456141	5.67E-08	2.31E-07
Potri.001G000500.1	0	1	1	nonDEG	0	1	1
Potri.001G000600.1	2.052966154	1	1	nonDEG	0	1	1
Potri.001G000700.1	-0.708800106	0.004397051	0.012765594	DEG	-0.57864149	0.017918689	0.034947571
Potri.001G000700.2	-0.708800191	0.004397867	0.012767455	DEG	-0.57864151	0.017920713	0.034949743
Potri.001G000800.1	-5.752295209	2.38E-32	1.19E-30	DEG	-2.07992598	3.60E-09	1.62E-08
Potri.001G000900.1	-0.106881527	0.677556136	0.863795048	nonDEG	-0.41485103	0.101230866	0.164720403
Potri.001G000900.2	-0.104177162	0.685100088	0.871321938	nonDEG	-0.39259678	0.12091177	0.192568953
Potri.001G000900.3	-0.104177155	0.685102882	0.871321938	nonDEG	-0.39259677	0.120918522	0.192575506
Potri.001G000900.4	-0.104177148	0.685105676	0.871321938	nonDEG	-0.39259677	0.120922395	0.192577474
Potri.001G000900.5	-0.104177141	0.685108471	0.871321938	nonDEG	-0.39259677	0.120926269	0.192579442
Potri.001G000900.6	-0.104177115	0.685118985	0.871321938	nonDEG	-0.39259676	0.12093309	0.192586104
Potri.001G001000.1	0	1	1	nonDEG	8.505002191	3.24E-16	2.35E-15
Potri.001G001100.1	2.856774447	0.51868682	0.711384175	nonDEG	7.42116742	1.57E-08	6.73E-08
Potri.001G001200.1	-4.268697283	0.026628012	0.061597157	nonDEG	-1.37868029	0.400219166	0.539694185
Potri.001G001200.2	-4.268697953	0.026625436	0.06159315	nonDEG	-1.37867193	0.400259753	0.539738949
Potri.001G001300.1	0.811016005	0.000942132	0.003226901	DEG	0.165737878	0.51788279	0.663092314

Good resource

<http://training.bioinformatics.ucdavis.edu/documentation/>

The screenshot shows a web browser window displaying the UC Davis Bioinformatics Core website. The page title is "UC Davis Bioinformatics Training Program" and the subtitle is "Bioinformatics courses, boot camps and workshops presented by the University of California, Davis Bioinformatics Core". The navigation menu includes "Home", "Documentation" (highlighted), "News", "Accommodations", "Galleries", "FAQ", and "Contact Us". The "Documentation" section is active, showing a list of courses and boot camps from December 2013, September 2013, and June 2013.

UC DAVIS Bioinformatics Core

Search

UC Davis Bioinformatics Training Program

Bioinformatics courses, boot camps and workshops presented by the University of California, Davis Bioinformatics Core

Home **Documentation** News Accommodations Galleries FAQ Contact Us

Documentation

Each course comes with its own documentation released after each course to the public. The documentation for our courses is below:

December 2013

- December 13, 2013 bootcamp: Introduction to the Amazon Cloud for Galaxy and the Command-Line
- December 12, 2013 bootcamp: Genome Assembly using Next Generation Sequence Data
- December 11, 2013 bootcamp: Next Generation Sequence Alignment and Variant Discovery
- December 10, 2013 bootcamp: Introduction to Next Generation Sequence Analysis with Galaxy

September 2013

- September 16-20, 2013: Bioinformatics Short Course 2013
- September 8-10, 2013: RNA-Seq Workshop: From Pipette to P-value!

June 2013

- June 21, 2013 bootcamp: Cloud Computing for Bioinformatics

Question and answer
