

Introduction to RNASeq

Malay (malay@uab.edu)

November 18, 2016

Contents

1	Introduction	1
2	Normalization of RNASeq data	1
3	Datasets	2
4	STAR	2
5	RSEM	2
5.1	Installing RSEM	2
5.2	Prepare reference	3
5.3	Calculate expression directly from STAR output	3
5.4	Simpler way to estimating expression	3
5.5	Differential expression	3
5.6	Volcano plot	3
6	EDGER	4
	Bibliography	5

1 Introduction

RNASeq is a very vast topic and tons of papers have been and are being written on the topic. The following is just an overview.

Originally the idea was proposed by Mortazavi *et al.* (2008). Although several modification of the original idea have been developed, the basics did not change. In this handout will use the latest in the RNASeq methodology through the use of software called RSEM (Li and Dewey, 2011).

2 Normalization of RNASeq data

People have proposed several methods of normalization of RNASeq data. For a comparison see Dillies *et al.* (2013).

3 Datasets

Every differential expression measurement should have biological replicates. For demonstration, we will use only 1 replicate for two biological conditions. But in real life, this should never be used. We will use two small datasets from Illumina Body Map project. These are samples prepared from adrenal gland and brain and only from chromosome 19. You can download the datasets here:

http://cmb.path.uab.edu/training/docs/CB2-201-2015/rnaseq_data.tar.gz

Unzip the file.

4 STAR

STAR is a modern fast aligner for RNASeq data to reference genome.

```
wget https://github.com/alexdobin/STAR/archive/2.5.1b.tar.gz
tar -xvzf 2.5.1b.tar.gz
cd STAR-2.5.1b
make
```

Put the software in your path

```
cd Linux_x86_64_static/
export PATH=$PATH:`pwd`
```

Prepare the reference genome:

```
mkdir hs
STAR --runThreadN 8 --genomeDir hs --runMode genomeGenerate \
    --genomeFastaFiles chr19.fa --sjdbGTFfile human_chr19.gtf
```

Now create the alignment. There is a special option for STAR to create a “transcriptome alignment” that could be fed directly to RSEM.

```
STAR --runThreadN 8 --genomeDir hs --readFilesIn adrenal_R1.fq \
    adrenal_R2.fq --quantMode TranscriptomeSAM
```

5 RSEM

RSEM is a cutting-edge RNASeq analysis package that is an end-to-end solution for differential expression, and simplifies the whole process. It also introduces a new more robust unit of RNASeq measurement called TPM.

5.1 Installing RSEM

```
wget http://deweylab.biostat.wisc.edu/rsem/src/rsem-1.2.19.tar.gz
tar -xvzf rsem-1.2.19.tar.gz
cd rsem-1.2.19/
```

```
make
export PATH=$PATH:`pwd`

# Install ebseq
module load R/R-3.1.2
make ebseq
cd EBSeq/
export PATH=$PATH:`pwd`
```

5.2 Prepare reference

```
rsem-prepare-reference --gtf human_chr19.gtf chr19.fa rsem/chr19
```

5.3 Calculate expression directly from STAR output

```
rsem-calculate-expression --no-bam-output --paired-end \
  --bam Aligned.toTranscriptome.out.bam rsem/chr19 adrenal
```

5.4 Simpler way to estimating expression

```
rsem-prepare-reference --gtf human_chr19.gtf --star --star-path \
  ../STAR-2.5.1b/bin/Linux_x86_64_static -p 8 chr19.fa hs/chr19
rsem-calculate-expression --paired-end --star --star-path \
  ../STAR-2.5.1b/bin/Linux_x86_64_static/ -p 8 adrenal_R1.fq \
  adrenal_R2.fq hs/chr19 adrenal_rsem
rsem-calculate-expression --paired-end --star --star-path \
  ../STAR-2.5.1b/bin/Linux_x86_64_static/ -p 8 brain_R1.fq brain_R2.fq \
  hs/chr19 brain_rsem
```

5.5 Differential expression

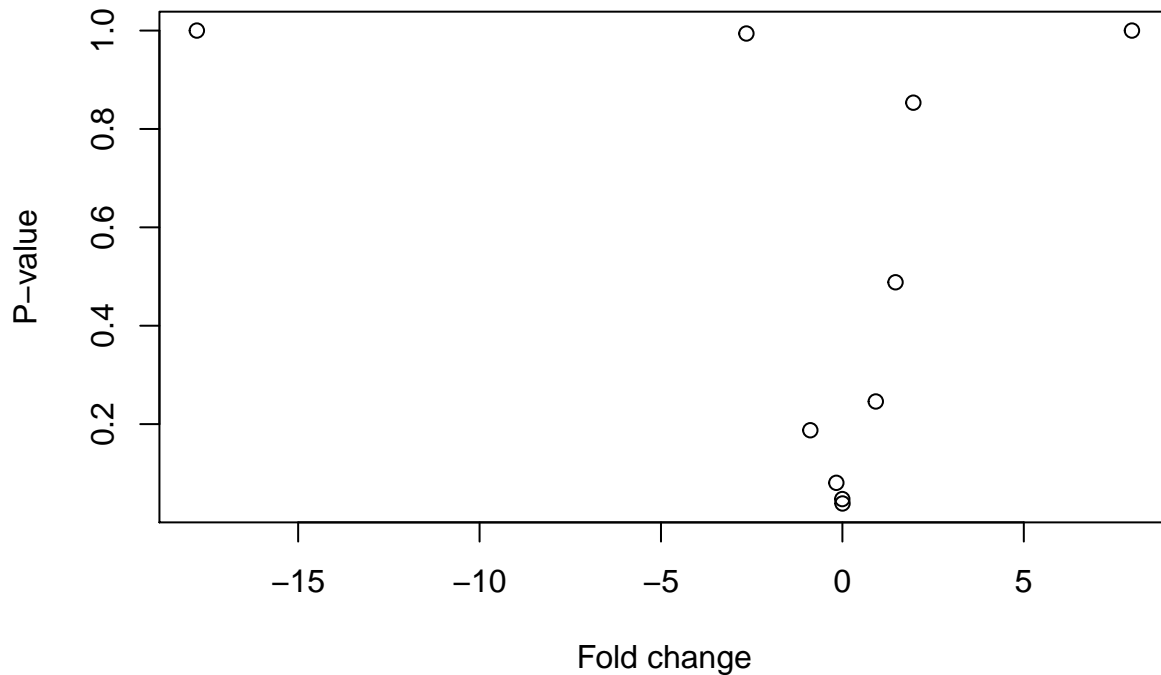
```
rsem-generate-data-matrix adrenal_chr19.genes.results human_chr19.genes.results \
  >diff-brain-adrenal.txt
rsem-run-ebseq diff-brain-adrenal.txt 1,1 expression.results.txt
rsem-control-fdr expression.results.txt 0.05 expression_final.txt
```

And we have our differentially expressed genes.

5.6 Volcano plot

Volcano plot is a good way to show the differentially expressed genes. For that we need the p-value for the differentially expressed genes and the fold change. Given by “PPEE” and “RealFC” values.

```
data<-read.table("expression.results.txt")
plot(log2(data$RealFC),data$PPDE,xlab="Fold change",ylab="P-value")
```



6 EDGER

```
raw.data <- read.table("../data/pnas_expression.txt",header=T)
head(raw.data)
counts <- raw.data[ , -c(1,ncol(raw.data))]
rownames(counts) <- raw.data$ensembl_ID
colnames(counts) <- paste(c(rep("C_R",4),rep("T_R",3)),c(1:4,1:3),sep="")

library(edgeR)
group <- c(rep("C", 4) , rep("T", 3))
cds <- DGEList( counts , group = group )
cds <- calcNormFactors(cds)
design <- model.matrix(~group)
y <- estimateDisp(cds, design)
fit <- glmQLFit(y,design)
qlf <- glmQLFTest(fit,coef=2)
topTags(qlf)
```

Bibliography

Dillies,M.-A. *et al.* (2013) A comprehensive evaluation of normalization methods for illumina high-throughput RNA sequencing data analysis. *Brief Bioinform*, **14**, 671–683.

Li,B. and Dewey,C.N. (2011) RSEM: Accurate transcript quantification from RNA-seq data with or without a reference genome. *BMC Bioinformatics*, **12**, 323.

Mortazavi,A. *et al.* (2008) Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nat. Methods*, **5**, 621–628.