RNAseq Walkthrough part 3/3

C. Ryan Campbell

Duke University

c.ryan.campbell@duke.edu

7 Nov 2017

Overview

1. tophat2 paths
2. htseq-count
3. DESeq2
   - cluster
   - RStudio
4. Running DESeq2

Today's Goals

- Check tophat2 paths
- Discuss htseq-count flags
- Format data for DESeq
- Run DESeq

tophat

- tophat2 - alignment software
- In: Sequence data
- Out: .bam file - where that data aligns to/fits in the genome

SLURM Interactive Node

```
srun -mem-per-cpu=4000MB -pty bash -i
```

- When you're done with an interactive node type `exit`
- Also, check your SLURM queue `squeue -u <netID>`
- If you're still running a "bash" job, use `scancel <job ID number>` to cancel it

tophat

- tophat2 is the command/software that aligns the reads to the genome
- It is already installed on the cluster
- So put the following location in your path, or in your bash_profile

```
export PATH=/opt/apps/tophat-bowtie/:$PATH
tophat2


export PATH=/opt/apps/tophat/bin/samtools/bin/:$PATH
samtools
```

tophat

tophat2 example format (all one line):

```
tophat2 -p <number of threads> -o <output dir> -G <gff file,
annotations> <bowtie2 index> <R1 fastq> <R2 fastq>
```

- Help can be found by running “tophat2”
- Or in the tophat2 manual online
- http://ccb.jhu.edu/software/tophat/manual.shtml

HTSeq

- python-based program to count reads
- Input:
- .bam file and .gtf/.gff
- Output:
- A table of counts by gene

htseq-count

- We'll be using htseq-count
- This will count the number of reads mapped to each gene
- That data will be taken into DESeq2

```
export PATH=/opt/apps/rhel7/Python-2.7.11/bin/:$PATH
```

htseq-count

- What are its flags and options?

```
htseq-count <options> <alignment bam> <gff file> > <count
output>
```

Files to Use

- I have some example files to use for the DESeq tutorial
- They're human RNAseq files from a hypoxia experiment:

```
ls -lthr /work/cc216/490S/cc216/RNAseq_pt3
```

Combining htseq Output

- There are four samples of htseq-count output:

```
s01.norm.counts, s02.norm.counts, s03.hypo.counts,
s04.hypo.counts
> head s01.norm.counts
3.8-1.4 0
3.8-1.5 0
5-HT3C2 0
A1BG 252
A1BG-AS1 47
```

Combining htseq Output

- Check that they're all the same length (have the same rows):

```
> wc -l s*counts
45381 s01.counts
45381 s01.norm.counts
45381 s02.counts
45381 s02.norm.counts
```

Combining htseq Output

- Use `paste` to combine these files:

```
> paste s01.norm.counts s02.norm.counts s03.hypo.counts
s04.hypo.counts
3.8-1.4 0 3.8-1.4 0 3.8-1.4 0 3.8-1.4 0
3.8-1.5 0 3.8-1.5 0 3.8-1.5 0 3.8-1.5 0
5-HT3C2 0 5-HT3C2 0 5-HT3C2 0 5-HT3C2 0
A1BG 252 A1BG 192 A1BG 175 A1BG 153
A1BG-AS1 47 A1BG-AS1 28 A1BG-AS1 31 A1BG-AS1 35
```

Combining htseq Output

- And `cut` to eliminate redundant columns:

```
> paste s01.norm.counts s02.norm.counts s03.hypo.counts
s04.hypo.counts | cut -f1,2,4,6,8
3.8-1.4 0 0 0 0
3.8-1.5 0 0 0 0
5-HT3C2 0 0 0 0
A1BG 252 192 175 153
A1BG-AS1 47 28 31 35
```

Finally write to a file:

```
> paste s01.norm.counts s02.norm.counts s03.hypo.counts
s04.hypo.counts | cut -f1,2,4,6,8 > hypoxia_hsap.counts
```

Move the count table down to your laptop

On your laptop's terminal (not slogin):

```
scp -v <yournetID>@dcc-slogin-02.oit.duke.edu:
/work/cc216/490S/cc216/RNAseq_pt3/*.counts
 /<your laptop location>/
```

(pay attention to the spaces! the spacing above isn't correct because the slide is too narrow)

DESeq2

- See the website for installation instructions
- Needs two things:
  1. A matrix of counts = “cts”
  2. A matrix of sample conditions = “coldata”

DESeq2

In R:

```
> head(cts, n = 4)
s01norm s02norm s03hypox s04hypox
3.8-1.4 0 0 0 0
3.8-1.5 0 0 0 0
5-HT3C2 0 0 0 0
A1BG 252 192 175 153

> coldata
condition type
s01norm "untreated" "paired-end"
s02norm "untreated" "paired-end"
s03hypox "treated" "paired-end"
s04hypox "treated" "paired-end"
```

DESeq2

- Order must match (01, 02, 03...)
- Name the columns and rows appropriately

```
> rownames(cts)
[1] "s01norm" "s02norm" "s03hypox" "s04hypox"


> colnames(cts)
[1] "condition" "type"
```

You use the same function to call (see above) define (see below) column and row names:

```
> colnames(coldata) <- c("condition","type")
```

DESeq2

- Get your data in this format
- Keep track of your work in a .Rmd file!!
- Then (and only then) proceed to running DESeq (see further slides)

Today's Goals

1. Check tophat2 paths
2. Discuss htseq-count flags
3. Format data for DESeq

DESeq2 guides

- Here are the DESeq guides that I have summarized in this walkthrough:
- Walkthrough Link
- Focus on “Quick Start” and more specifically:
- Setting the R objects `cts` and `coldata` correctly
- Using `paste` (a unix command) to format your data into `cts`

http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html

DESeq2

- To run DESeq, first create a “DESeq dataset”

```
> dds <- DESeqDataSetFromMatrix(countData = cts,
colData = coldata,
design =   condition)

> dds
```

Where cts and coldata are the files described earlier

Output:

```
class:  DESeqDataSet
dim:  45381 4
metadata(1):  version
```

DESeq2

- Let's throw out genes with low expression levels (here, below 10)

```
> keep <- rowSums(counts(dds)) >= 10
> dds <- dds[keep,]
> dds$condition <- relevel(dds$condition, ref = "untreated")
> dds
```

Output:

```
class:  DESeqDataSet
dim:  17380 4
metadata(1):  version
```

We've shrunk our gene list by roughly 60 percent

DESeq2

- Now... Run DESeq!!!

```
> dds <- DESeq(dds)
> res <- results(dds)
> res
```

Output:

```
log2 fold change (MAP): condition treated vs untreated
Wald test p-value:  condition treated vs untreated
DataFrame with 17380 rows and 6 columns
baseMean log2FoldChange lfcSE stat pvalue

A1BG 189.927175 -0.20541335 0.16773404 -1.22463727 0.2207119
A1BG-AS1 34.902293 0.02668120 0.26939459 0.09904134 0.9211054
```

DESeq2

- A few final things to do and explore:
- Sort the results by p-value

```
> resOrdered <- res[order(res$pvalue),]
```

- Summarize the results

```
> summary(res)
```

- How many genes are significant (at 0.10 and 0.05)?

```
> sum(res$padj < 0.1, na.rm=TRUE)
> sum(res$padj < 0.05, na.rm=TRUE)
```

The End