A Tutorial: Genome-based RNA-Seq Analysis Using the TUXEDO Package

The following details the steps involved in:

- Aligning RNA-Seq reads to a genome using Tophat
- Assembling transcript structures from read alignments using Cufflinks
- Visualizing reads and transcript structures using IGV
- Performing differential expression analysis using Cuffdiff
- Expression analysis using CummeRbund

All required software and data are provided pre-installed on a VirtualBox image. See companion 'Rnaseq_Workshop_VM_installation.pdf' for details. Data content and environment configurations are described therein and referenced below.

Before Running:

After installing the VM, be sure to quickly update the contents of the rnaseq_workshop_data directory by:

```
% cd rnaseq_workshop_data

% svn up
```

This way, you'll have the latest content, including any recent bugfixes.

Automated and Interactive Execution of Activities

To avoid having to cut/paste the numerous commands shown below into a unix terminal, the VM includes a script 'runTrinityDemo.pl' that enables you to run each of the steps interactively. To begin, simply run:

```
% runTuxedoDemo.pl -I --DE
```

The -I parameter indicates to run interactively, and --DE indicates to include the differential expression analysis activities.

Use Tophat and Cufflinks to align reads and assemble transcripts

a. process condition A reads

run Tophat to generate alignments for condition A reads
% tophat -I 1000 -i 20 -o condA_tophat_out genome condA.left.fa condA.right.fa

index the alignment bam file for use by downstream tools including visualization
% samtools index condA_tophat_out/accepted_hits.bam

generate transcript structures using Cufflinks
% cufflinks -o condA_cufflinks_out condA_tophat_out/accepted_hits.bam

b. process condition B reads

run Tophat to generate alignments for condition B reads
% tophat -I 1000 -i 20 -o condB_tophat_out genome condB.left.fa condB.right.fa

index the resulting bam file
% samtools index condB_tophat_out/accepted_hits.bam

generate transcript structures using cufflinks
% cufflinks -o condB_cufflinks_out condB_tophat_out/accepted_hits.bam

Merge separately assembled transcript structures into a cohesive set:

First, create a file that lists the names of the files containing the separately reconstructed transcripts, which can be done like so:

first writes the file
% echo condA_cufflinks_out/transcripts.gtf > assemblies.txt

writes in append mode to add the second filename
% echo condB_cufflinks_out/transcripts.gtf >> assemblies.txt

verify that this file now contains both filenames:
% cat assemblies.txt
condA_cufflinks_out/transcripts.gtf
condB_cufflinks_out/transcripts.gtf

And now we're ready to merge the transcripts using cuffmerge:

% cuffmerge -s genome.fa assemblies.txt

The merged set of transcripts should now exist as file "merged_asm/merged.gtf'.

View the reconstructed transcripts and the tophat alignments in IGV

% java -jar $IGV/igv.jar -g `pwd`/genome.fa `pwd`/merged_asm/merged.gtf,`pwd`/genes.bed,`pwd`/condA_tophat_out/accepted_hits.bam,`pwd`/condB_tophat_out/accepted_hits.bam

Pan the genome, examine the alignments, known genes and reconstructed genes.

Do the alignments agree with the known gene structures (ex. Intron placements)?

Do the cufflinks-reconstructed transcripts well represent the alignments?

Do the cufflinks-reconstructed transcripts match the structures of the known transcripts?

Differential expression analysis using cuffdiff and cummeRbund:

% cuffdiff -o diff_out -b genome.fa -L condA,condB -u merged_asm/merged.gtf condA_tophat_out/accepted_hits.bam condB_tophat_out/accepted_hits.bam

Examine the output files generated in the diff_out/ directory.

A table containing the results from the gene-level differential expression analysis can be found as 'diff_out/gene_exp.diff'. Examine the top lines of this file like so:

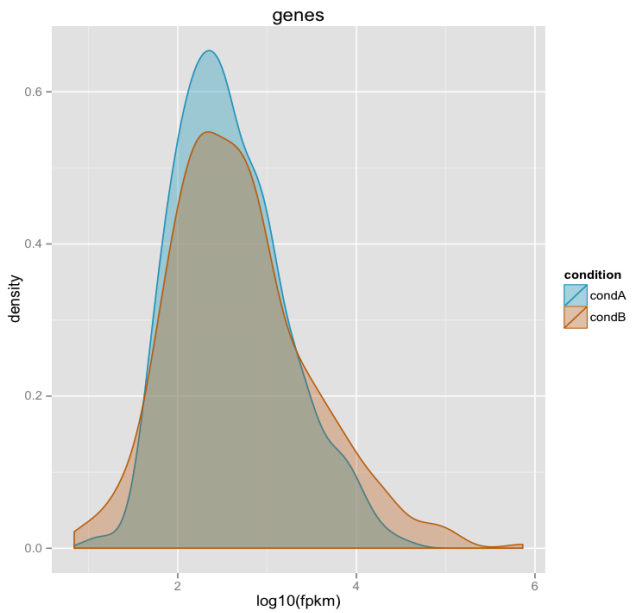
% head diff_out/gene_exp.diff

Use 'cummeRbund' to analyze the results from cuffdiff:

% R
(note, to exit R, type cntrl-D, or type "q()").

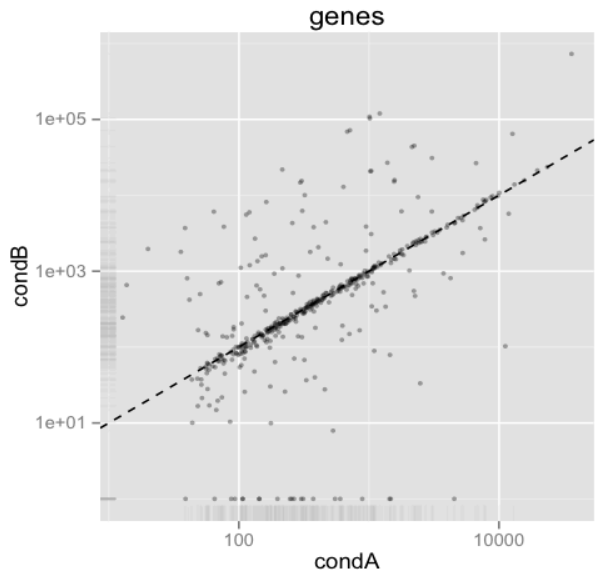
> **Optional: To automate running of the steps below interactively, you can do the following:**
>
> **> source("cummeRbund.demo.R")**
>
> **and then follow along below.**

```
# load the cummerbund library into the R session
> library(cummeRbund)

# import the cuffdiff results
>cuff = readCufflinks('diff_out')

# examine the distribution of expression values for the reconstructed transcripts
>csDensity(genes(cuff))
```



Examine transcript expression values in a scatter plot
Expression values are typically log-normally distributed. This is just a sanity check.

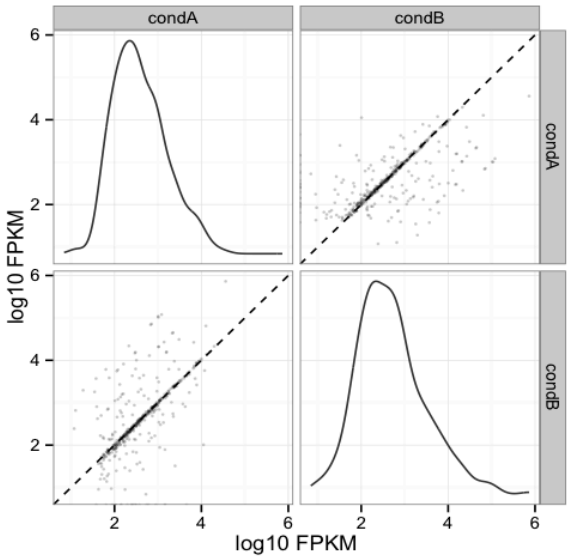
>csScatter(genes(cuff), 'condA', 'condB')



Strongly differentially expressed transcripts should fall far from the linear regression line.

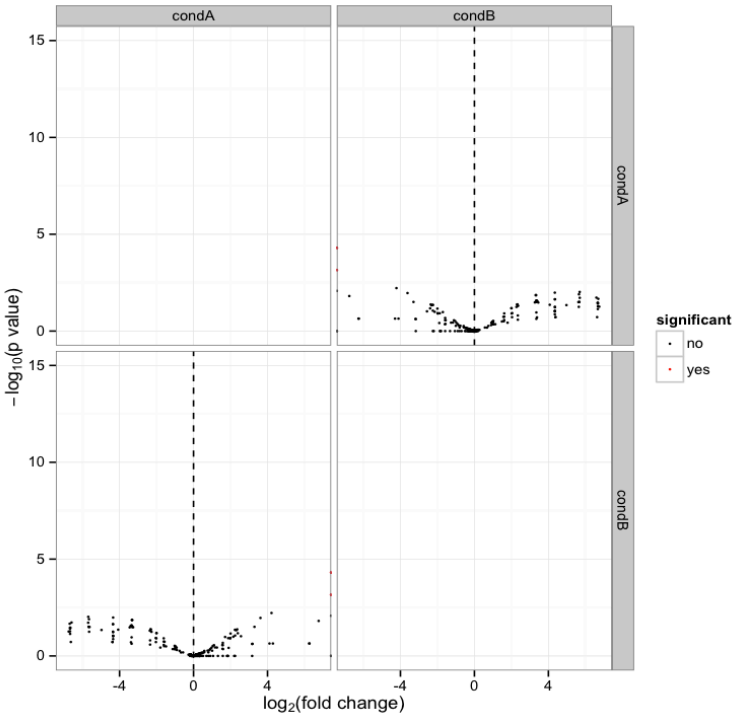
Examine individual densities and pairwise scatterplots together.

> csScatterMatrix(genes(cuff))



Volcano plots are useful for identifying genes most significantly differentially expressed.

> csVolcanoMatrix(genes(cuff), 'condA', 'condB')



Extract the 'genes' that are significantly differentially expressed (red points above)

retrieve the gene-level differential expression data
> gene_diff_data = diffData(genes(cuff))

how many 'genes'?
> nrow(gene_diff_data)

from the gene-level differential expression data, extract those that
are labeled as significantly different.
note, normally just set criteria as "significant='yes'", but we're adding an
additional p_value filter just to capture some additional transcripts for
demonstration purposes only. This simulated data is overly sparse and actually
suboptimal for this demonstration (in hindsight).

>sig_gene_data = subset(gene_diff_data,(significant=='yes' | p_value < 0.1))

how many?
> nrow(sig_gene_data)

Examine the entries at the top of the unsorted data table:

> head(sig_gene_data)

```
gene_id sample_1 sample_2 status  value_1   value_2 log2_fold_change
4  XLOC_000004    condA    condB     OK  307.128     0.000             -Inf
8  XLOC_000008    condA    condB     OK  266.134     0.000             -Inf
11 XLOC_000011    condA    condB     OK  322.349 10143.700          4.97582
15 XLOC_000015    condA    condB     OK  199.150     0.000             -Inf
17 XLOC_000017    condA    condB     OK 4317.350   821.552         -2.39372
22 XLOC_000022    condA    condB     OK  134.732   650.882          2.27230
   test_stat p_value    q_value significant
4         NA 0.00005 0.00243125         yes
8         NA 0.00005 0.00243125         yes
11   3.16109 0.04565 0.34149700          no
15        NA 0.00070 0.01945000         yes
17  -2.57482 0.06720 0.40845000          no
22   2.15938 0.05690 0.36890200          no
```

You can write the list of significantly differentially expressed genes to a file like so:

> write.table(sig_gene_data, 'sig_diff_genes.txt', sep = '\t', quote = F)

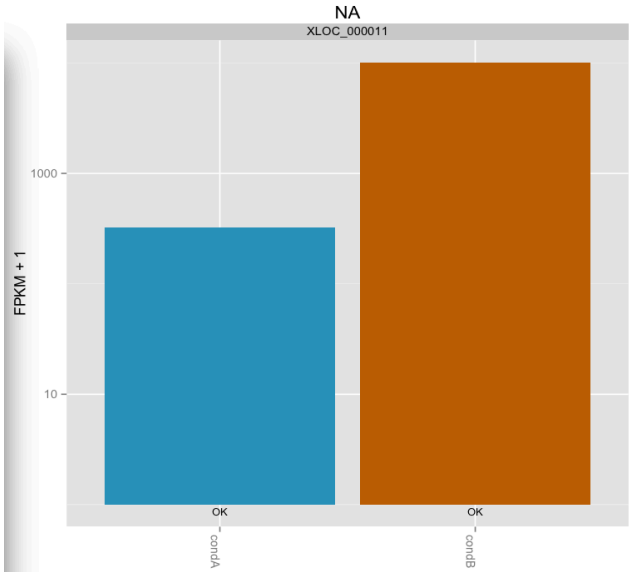
examine the expression values for one of your genes that's diff. expressed:

select expression info for the one gene by its gene identifier:
(note we're naming the variable the same as the
transcript name, so don't be confused by this)

> var_XLOC_000011 = getGene(cuff, 'XLOC_000011')

now plot the expression values for the gene under each condition
(error bars are only turned off here because this data set is both simulated
and hugely underpowered to have reasonable confidence levels)

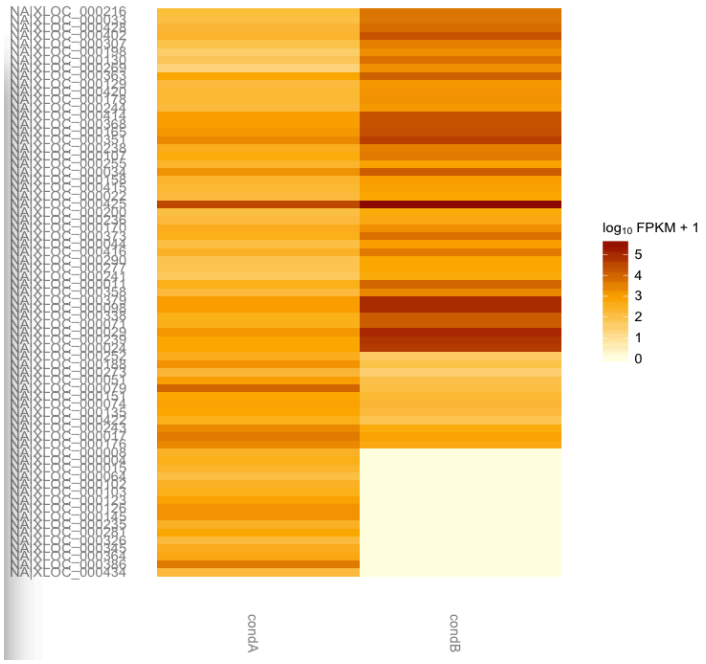
> expressionBarplot(var_XLOC_000011, logMode=T, showErrorbars=F)



Draw a heatmap showing the differentially expressed genes

first retrieve the 'genes' from the 'cuff' data set by providing a
a list of gene identifiers like so:
>sig_genes = getGenes(cuff, sig_gene_data$gene_id)

# now draw the heatmap
csHeatmap(sig_genes, cluster='both')



More information on using the Tuxedo package can be found at:

Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, Pimentel H, Salzberg SL, Rinn JL, Pachter L.
Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. Nat Protoc. 2012 Mar 1;7(3):562-78. doi: 10.1038/nprot.2012.016.
http://www.nature.com/nprot/journal/v7/n3/full/nprot.2012.016.html

The CummeRbund manual:
http://compbio.mit.edu/cummeRbund/manual_2_0.html

(note, most of the tutorial provided here is based on the above two resources)

and the Tuxedo tool websites:

TopHat: http://tophat.cbcb.umd.edu/
Cufflinks: http://cufflinks.cbcb.umd.edu/
CummeRbund: http://compbio.mit.edu/cummeRbund/