Khan Inan
Week 10 assignment
BI-GY 7653

Q1.1. A key consideration when conducting any type of RNA-seq analysis is the percentage of reads that aligned. The Salmon approach may have a low mapping rate if the kmer length is too stringent (kmer matches between read and transcript are too long).

Review the logging output of Salmon. Report the mapping rates for each sample produced by Salmon. Do you consider these to be too low? If so, how might you re-run Salmon to increase the mapping rates (see Salmon website above)? [1 point]

```
PDAC253= 8.71844%
PDAC266=11.1517%
PDAC273=22.8452%
PDAC282=15.9158%
PDAC286=11.8775%
PDAC306=8.93615%
PDAC316=17.2182%
PDAC318=12.1166%

# Since the mapping rate for all 8 samples seem to be at the lower end (between 8%-23%), I would consider these to be too lo
w which is likely due to kmer lengths to be too stringent. To re-run Salmon with the goal of increasing mapping rate by choo
sing a smaller k-mer length, we will need to re-run the salmon indexer using the script:

salmon index -t Pdac_Barhee_chr_unan_180126.all.maker.transcripts_HC_shuffled_normalized.fa -i Pdac_Barhee_chr_unan_180126.a
ll.maker.transcripts_HC_shuffled_normalized_index -k 31

#If we were to re-run the salmon at a different kmer length,then we would need to create a new index using the script above
and specify a different, smaller k value. After, we can re-run Salmon using the new index file.
```

Q1.2. Choose a sample and review the output file “quant.sf”. What are the columns in the output? Please provide an explanation of each column (see “Salmon Output File Formats” in “Output” section of documentation page referenced above).

The column names of the output are Name, Length, effectivelength, TPM and NumReads. The first column identifies the name of the transcript that is the target, taken from the transcript database file from the input. The column named length specifies the number/length of the total amount of nucleotides. The column labeled effective length is the length of the transcripts after taking into account variables such as GC fragment bias and fragment length distribution. The column labeled TPM is just an estimate of the presence of each kind of transcript in the TPM unit or transcripts per million. Lastly the numreads column in an estimate of the number of redas that are mapped to each transcript

Q1.3a. What library type did Salmon infer for the input reads?

a. **IU (“inward”, “unstranded”)**
b. IS (“inward”, “stranded”)
c. OU (“outward”,“unstranded”)
d. OS (“outward”,“stranded”)
e. all of the above

Q1.3b What library type do you think Salmon would have inferred if the date palm RNA-seq had been a stranded library preparation? Please use the two-letter syntax reported in the Salmon documentation.

a. **IU (“inward”, “unstranded”)**
b. IS (“inward”, “stranded”)
c. OU (“outward”,“unstranded”)
d. OS (“outward”,“stranded”)
e. all of the above

Q1.3c Explain in your own words what is the difference between a stranded (=“strand-specific”) and unstranded library?

A stranded library is one where we are sure of the orientation of the reads. What this essentially means is that we know which strand (the + or -) of the DNA that the specific read is from. In an unstranded library, it is the exact opposite where we don’t know which strand the read is from. In terms of preparation, the stranded library would be a bit more challenging because we need to label + or - strands of cDNA in order to be able to tell the difference

Q1.3d Which do you think is typically preferred for performing DGE analysis? Why?

I would assume that stranded would be preferred simply because RNA and genes can overlap and have a lot of similarities with its paired strand, and it can be difficult to determine which strand it is from. As a result if we use the unstranded library then we are going to have issues with determining gene expression levels, so stranded libraries are preferred.

Q2.1. A nice property of TPMs is that TPMs for alternative transcripts can be added to get the TPM value for all transcripts in a gene. Why does DESeq2 need to convert TPMs to gene counts instead of just using the gene-level TPMs directly in the statistical analysis [1 point]?

Normal count methods for differential gene expression have resulted in many false positive because the difference in transcript usage. An example would be that there are genes that don’t have a different amount of mRNA but there have a different combination of isoforms. Salmon solves this issue because it can estimate the presence of transcripts and their abundances, and also find the sum of TPMs and also convert the calculated values into counts to then perform standard count gene analysis. Also using tools like tximport, bioinformaticians can use normal count methods for gene analysis while also avoiding differences in transcript usage

Q3.1. Please report the commands you executed to complete your analysis [1 point].

```
directory <- "/Users/khan/Desktop/ngs.week10"

library(tximport)
sample_names <- c('PDAC253','PDAC282','PDAC286','PDAC316','PDAC266','PDAC273','PDAC306','PDAC318')
sample_condition <- c(rep('highSucrose',4),rep('lowSucrose',4))


files <- file.path("/Users/khan/Desktop/ngs.week10",sample_names,paste(sample_names,".transcripts_quant",sep=""),'quant.s
f')
names(files) <- sample_names

file.exists("/Users/khan/Desktop/ngs.week10/PDAC253/PDAC253.transcripts_quant/quant.sf")

tx2gene <- read.table("/Users/khan/Desktop/ngs.week10/Pdac_Barhee_chr_unan_180126_maker_HC.tx2gene",header=F,sep=",")

txi <- tximport(files, type="salmon", tx2gene=tx2gene)

samples <- data.frame(sample_names=sample_names,condition=sample_condition)
row.names(samples) <- sample_names


library("DESeq2")
ddsTxi <- DESeqDataSetFromTximport(txi,
                                   colData = samples,
                                   design = ~ condition)
class(ddsTxi)

ddsTxi

keep <- rowSums(counts(ddsTxi)) >= 10
dds <- ddsTxi[keep,]

dds <- DESeq(dds)
class(dds)

res.shrunk <- lfcShrink(dds, contrast = c('condition','lowSucrose','highSucrose'), type= "ashr" )
res.shrunkOrdered <- res.shrunk[order(res.shrunk$pvalue),]

res.shrunkOrdered

library(ggplot2)
ggplot(as.data.frame(res.shrunk),aes(pvalue)) + geom_histogram(fill="light blue",color='black')
```

Q3.2. In Week 9 you considered the problem of multiple comparisons and application of FDR. Now Lets take a more careful look at the uncorrected p-value distribution which can alert us to potential problems in the statistical analysis.

A histogram is a way of visualizing the density distribution of numeric values (such as p-values) that are binned typically in evenly spaced intervals. Construct a histogram of the raw (uncorrected) p-values from the DESeqResults object. Note, you may need to change the “res” variable in the command below based on the value you assigned to the output of the lfcshrink function (“res” variable is a DESeqResults object).

ggplot(as.data.frame(res),aes(pvalue)) + geom_histogram(fill="light blue",color='black')

Which pattern below (a-e) do you observe in your histogram? What does this suggest about the impact of sugar composition on gene expression? Please select the single best answer and include your p-value histogram in your answer. [1 point].

a. a uniform distribution (no large “peaks” or “valleys” in the distribution) suggests a disappointing result that very few if any genes are differentialy expressed in the analysis
b. depletion of values with low p-values. This could suggest a confounding factor such as a batch effect.
c. a multi-modal distribution (multiple humps) could indicate a strong correlation structure in the data and may suggest that the statistical methods applied are inappropriate
d. <u>an enrichment of low p-values. This is the expected result if there is a large class of differentially expressed genes between treatment and control.</u>
e. a choppy appearance (some p-values observed, others not observed) could mean the P-values are fall into discrete classes (undesirable)

Q3.3. Report the results table for the top 10 differentially expressed genes according to adjusted p-value (i.e., FDR). Then, for each of the three genes below Do you consider the gene(s) to be differentially expressed, both statistically and biologically in terms of log fold-change? Why or why not [1 point].

Pdac_HC_chr14G0022900 (cell wall invertase enzyme)
Pdac_HC_chr14G0023100 (cell wall invertase enzyme)
Pdac_HC_chr14G0028200 (alkaline/neutral invertase enzyme)

```
log2 fold change (MMSE): condition lowSucrose vs highSucrose
Wald test p-value: condition lowSucrose vs highSucrose
DataFrame with 3 rows and 5 columns
                      baseMean log2FoldChange     lfcSE      pvalue        padj
                     <numeric>      <numeric> <numeric>   <numeric>   <numeric>
Pdac_HC_chr14G0022900  8502.970       9.692887  0.607679 5.62630e-60 9.79145e-56
Pdac_HC_chr14G0023100  6923.473       5.869339  0.650034 1.23136e-21 5.35735e-18
Pdac_HC_chr14G0028200   582.807       0.364476  0.216343 4.88953e-02 2.26907e-01

# I would consider both Pdac_HC_chr14G0023100 and Pdac_HC_chr14G0022900 genes to be expressed differentially since they have
a log2 FC value greater than 1 which is high. This means that the statistically differentially expressed gene shows at least
a two-fold change in expression. Both genes also have a low padj value which is the adjusted p-value (the FDR). Both also pa
ssed the FDR threshold which I set as 0.01 which is on the more strict side. A FDR threshold that I put at 0.01 means that
1% or less of these genes are expected to be false positives. Since both Pdac_HC_chr14G0023100 and Pdac_HC_chr14G0022900 hav
e a padj value of greater than 0.01 with high log2FoldChange (at least greater than 1), I would only consider these two gene
s be differentially expressed, both statistically and biologically in terms of log fold-change. I would not consider Pdac_HC
_chr14G0028200 to be differentially expressed as it does not have a log2FoldChange of at least 1 and did not pass the FDC th
reshold.
```

Q3.4. Show your code and plots.

Then, read the DESeq2 vignettesection “Dispersion plot and fitting alternatives”. Then read this post and the answer by Michael Love, the lead developer of DESeq2:

https://support.bioconductor.org/p/63244/

What is the relationship of genewise dispersions and the mean of normalized counts in each of your plots? Do you see a difference among the three methods? Should the fitType be changed from the default “parametric” method as run by the DESeq wrapper function? [1 point]

```
dds <- estimateDispersions(dds, fitType="parametric")
plotDispEsts(dds)

dds <- estimateDispersions(dds, fitType="local")
plotDispEsts(dds)

dds <- estimateDispersions(dds, fitType="mean")
plotDispEsts(dds)
```

Based on the plot that was created from the parametric fit, the curve seems to fit the observed dispersion relationship. The curve created from the local fit, however, does not fit this same relationship. I see a difference in the three methods. The mean fit methods is not a good fit, so as a result fitType shouldn’t be changes since the default does seem to fit the relationship the best out of the three

Q3.5. In Week 8 and 9, you ran DGE analysis with an exon union approach and this week rans DGE using a Salmon + tximport + DESeq2 workflow. What are three reasons why the Salmon + tximport + DESeq2 workflow may be preferred over the STAR + htseq-count + DESeq2 workflow? [1 point]

Salmon + tximport + DESeq2 would be preferred over STAR + htseq-count + DESeq2 because

1. **Using the transcript presence estimation allows us to correct changes in the gene length across the samples and we can avoid false positive as a result**
2. **The tools use less memory and are generally faster and more efficient**
3. **It increases the sensitivity since it allows us to align multiple genes that have homologous fragments and we don’t have to discard them**