RNASeq-Pasilla

Daniel Scheuermann

2024-01-31

# Intro

This script and analysis is meant to mimic a study done in 2010 ([link](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3032923/)). The study consisted of studying the Pasilla gene using RNA-seq data. The Pasilla gene was depleted before the total RNA was isolated to prepare single-end and paired-end libraries for the treated vs untreated samples. From there, the libraries were sequenced to obtain the RNA-seq data for each specific sample. We will be comparing the treated vs untreated samples in this script.

## Data Setup

Here you can see the data set directly split into the treated and untreated samples.

The two separate samples can be compared to show the effects that ‘Pasilla’ gene depletion has on gene expression.

The factors are single vs paired samples and treated vs untreated samples.

count\_data <- read.csv("data/count\_matrix.csv", header = TRUE, row.names = 1)  
  
# colnames(count\_data)  
# head(count\_data)  
  
# sample information  
  
sample\_info <- read.csv("data/design.csv", header = TRUE, row.names = 1)  
  
# colnames(sample\_info)  
head(sample\_info)

## Treatment Sequencing  
## GSM461176\_untreated\_single untreated SE  
## GSM461177\_untreated\_paired untreated PE  
## GSM461178\_untreated\_paired untreated PE  
## GSM461179\_treated\_single treated SE  
## GSM461180\_treated\_paired treated PE  
## GSM461181\_treated\_paired treated PE

# setting factor levels  
sample\_info$Treatment <- factor(sample\_info$Treatment)  
sample\_info$Sequencing <- factor(sample\_info$Sequencing)

## DESeq2 Setup

In this section, we are just creating the DESeq2 object and specifying the treatment factors.

# creating the deseq object  
dds <- DESeqDataSetFromMatrix(countData = count\_data, colData = sample\_info, design = ~Sequencing + Treatment)  
  
# treatment factor reference  
dds$Treatment <- factor(dds$Treatment, level = c("untreated","treated"))

## Data Manipulation

We make the data easier to work with by removing genes who have a count less than 10. From there, we can perform a DESeq2 analysis to identify the differentially expressed genes. Note the top 10 values. We can tell they are differentially expressed by look at the pvalues.

# filtering the genes  
  
# we are looking at genes whose count is greater than 10  
keep <- rowSums(counts(dds)>10) >= min(table(sample\_info$Treatment))   
  
dds <- dds[keep,]  
  
# identifying diferentially expressed genes  
dds <- DESeq(dds)

## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing

deseq\_results <- results(dds)  
  
# transfer the deseq results into a data frame  
deseq\_results <- as.data.frame(deseq\_results)  
  
# ordering  
deseq\_results <- deseq\_results[order(deseq\_results$pvalue),]  
  
head(deseq\_results)

## baseMean log2FoldChange lfcSE stat pvalue  
## FBgn0039155 1087.088 -4.599523 0.15245724 -30.16927 5.994855e-200  
## FBgn0003360 6410.646 -3.164152 0.10938392 -28.92703 5.459703e-184  
## FBgn0026562 65125.951 -2.465742 0.08754260 -28.16619 1.518038e-174  
## FBgn0025111 2194.660 2.831895 0.10143424 27.91853 1.589601e-171  
## FBgn0029167 5432.388 -2.190908 0.09756613 -22.45562 1.127744e-111  
## FBgn0035085 928.391 -2.584359 0.12147633 -21.27459 1.952308e-100  
## padj  
## FBgn0039155 4.997911e-196  
## FBgn0003360 2.275877e-180  
## FBgn0026562 4.218628e-171  
## FBgn0025111 3.313125e-168  
## FBgn0029167 1.880401e-108  
## FBgn0035085 2.712732e-97

# count(deseq\_results)

## Looking at specific genes

By grabbing the results of specific genes from the data frame produced, we can clearly see the Pasilla gene is down-regulated by the RNA treatment due to it’s low p-value and negative log2FoldChange value.

# looking specifically at FBgn0039155  
deseq\_results["FBgn0039155",]

## baseMean log2FoldChange lfcSE stat pvalue  
## FBgn0039155 1087.088 -4.599523 0.1524572 -30.16927 5.994855e-200  
## padj  
## FBgn0039155 4.997911e-196

# testing whether or not the Pasilla gene is downregulated by the RNA treatment.  
deseq\_results["FBgn0261552",]

## baseMean log2FoldChange lfcSE stat pvalue  
## FBgn0261552 8915.949 -1.780947 0.1510969 -11.78679 4.566333e-32  
## padj  
## FBgn0261552 1.312742e-29

# Data Manipulation 2

After filtering the data and only selecting the differentially expressed genes, we can see that there’s a total of 234 differentially expressed genes.

# filtering  
  
# selecting the differentially expressed genes (alpha = 0.05)  
filtered <- deseq\_results %>% filter(deseq\_results$padj < 0.05)  
  
filtered <- filtered %>% filter(abs(filtered$log2FoldChange) > 1)  
  
# head(filtered)  
  
# number of differentially expressed genes  
dim(filtered)

## [1] 234 6

## Storing Results

Check the data folder for the data frames produce and normalized counts.

# save manipulated data in new csv files  
write.csv(deseq\_results, 'data/deseq\_result.all.csv')  
write.csv(filtered, 'data/deseq\_result.filtered.csv')  
  
normalized\_counts <- counts(dds, normalized = TRUE)  
head(normalized\_counts)

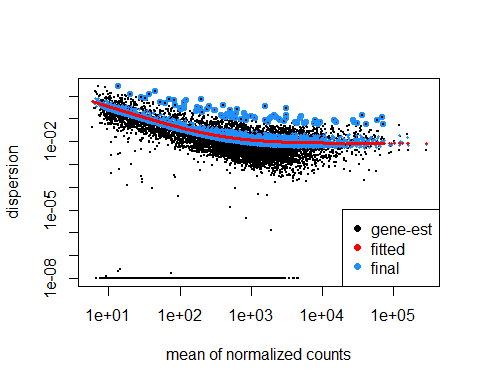
## GSM461176\_untreated\_single GSM461177\_untreated\_paired  
## FBgn0010247 508.18787 400.861508  
## FBgn0086378 524.41447 493.188211  
## FBgn0263977 1400.65132 904.567951  
## FBgn0069923 20.65205 5.843462  
## FBgn0039955 901.31433 666.154692  
## FBgn0259821 191.03143 176.472559  
## GSM461178\_untreated\_paired GSM461179\_treated\_single  
## FBgn0010247 324.13248 994.23589  
## FBgn0086378 590.26230 1075.84641  
## FBgn0263977 956.47513 1922.55559  
## FBgn0069923 12.51038 42.37469  
## FBgn0039955 547.04464 1081.33943  
## FBgn0259821 142.16337 438.65656  
## GSM461180\_treated\_paired GSM461181\_treated\_paired  
## FBgn0010247 439.0813 291.88658  
## FBgn0086378 585.8151 453.84245  
## FBgn0263977 1064.1001 823.50445  
## FBgn0069923 20.1619 13.72507  
## FBgn0039955 518.6088 426.39231  
## FBgn0259821 348.3528 302.86664  
## GSM461182\_untreated\_single  
## FBgn0010247 822.28940  
## FBgn0086378 1031.22178  
## FBgn0263977 1290.24905  
## FBgn0069923 28.10202  
## FBgn0039955 1064.21110  
## FBgn0259821 200.37959

write.csv(normalized\_counts, 'data/normalized\_counts.csv')

# Visualization (Dispersion and PCA)

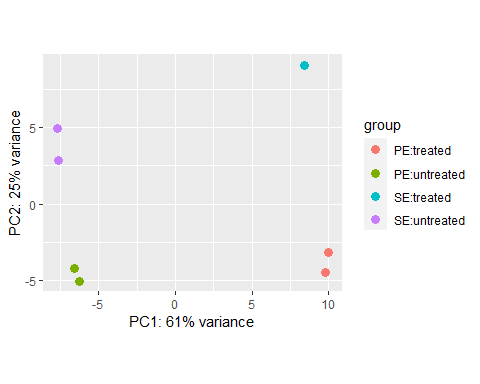
There’s a clear separation between treated and untreated samples. this demonstrated that there’s a path in the data showing a difference in treated vs untreated data samples. The same is true for pairwise vs single samples.

# dispersion plot  
plotDispEsts(dds)



# principal component analysis plot  
  
# variance stabilization  
vsd <- vst(dds, blind = FALSE)  
  
# pca plot  
plotPCA(vsd, intgroup = c("Sequencing", "Treatment"))

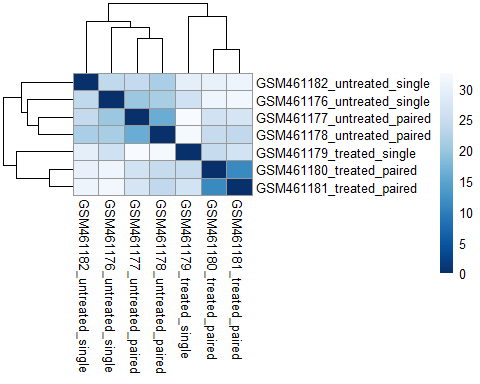
## using ntop=500 top features by variance



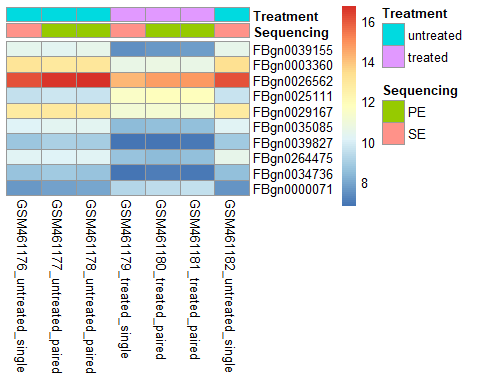
# Visualization (HeatMaps and Volcano)

Based on the heatmaps, it seems like GSM461180\_treated\_paired sample is similar to GSM461181\_treated\_pairs sample. Meanwhile, the rest of the samples seem to be relatively different.

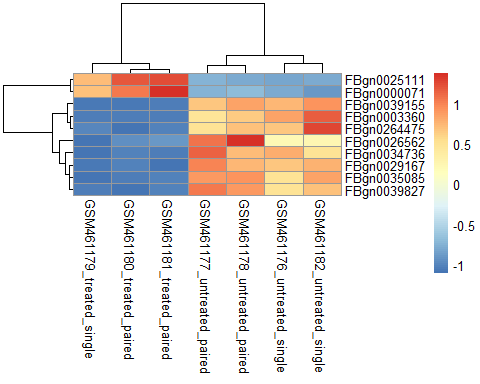
# generate distance matrix  
sample\_dists <- dist(t(assay(vsd)))  
sample\_dists\_matrix <- as.matrix(sample\_dists)  
# colnames(sample\_dists\_matrix)  
  
# choose color scheme  
colors <- colorRampPalette(rev(brewer.pal(9,"Blues")))(255)  
  
# generated the heat map  
pheatmap(sample\_dists\_matrix, clustering\_distance\_rows = sample\_dists, clustering\_distance\_cols = sample\_dists, color = colors)



# heatmap of top 10 log transformed normalized counts  
# sort by the log transformed normalized counts and select first 10 values  
top\_hits <- deseq\_results[order(deseq\_results$padj),][1:10,]  
top\_hits <- row.names(top\_hits)  
# top\_hits  
write.csv(top\_hits, "data/top\_hits.csv")  
  
# log transformation  
rld <- rlog(dds, blind = FALSE)  
  
# heatmap without clustering  
# pheatmap(assay(rld)[top\_hits,], cluster\_rows = FALSE, show\_rownames = TRUE, cluster\_cols = FALSE)  
  
# heatmap with clustering  
# pheatmap(assay(rld)[top\_hits,])  
  
# adding annotations  
annot\_info <- as.data.frame(colData(dds)[,c("Sequencing","Treatment")])  
  
# heatmap with annotations  
pheatmap(assay(rld)[top\_hits,], cluster\_rows = FALSE, show\_rownames = TRUE, cluster\_cols = FALSE, annotation\_col = annot\_info)



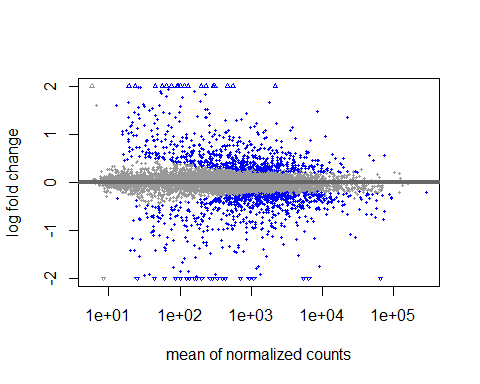
# heatmap of z-scores  
  
# calculating z-scores  
cal\_z\_score <- function(x) {(x-mean(x)) / sd(x)}  
  
z\_scores <- t(apply(normalized\_counts, 1, cal\_z\_score))  
z\_score\_subset <- z\_scores[top\_hits,]  
pheatmap(z\_score\_subset)



# MA plot  
  
# plotMA(dds, ylim = c(-2,2))  
  
# removing noise  
resLFC <- lfcShrink(dds, coef = "Treatment\_treated\_vs\_untreated", type = "apeglm")

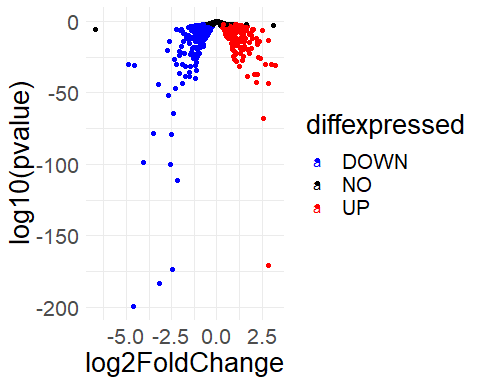
## using 'apeglm' for LFC shrinkage. If used in published research, please cite:  
## Zhu, A., Ibrahim, J.G., Love, M.I. (2018) Heavy-tailed prior distributions for  
## sequence count data: removing the noise and preserving large differences.  
## Bioinformatics. https://doi.org/10.1093/bioinformatics/bty895

plotMA(resLFC, ylim = c(-2,2))



# Volcano plot  
# changed the resLFC to a data frame to make a plot with little noise  
resLFC <- as.data.frame(resLFC)  
  
# labeling genes (diferentially expressed vs otherwise)  
resLFC$diffexpressed <- "NO"  
resLFC$diffexpressed[resLFC$log2FoldChange > 0.1 & resLFC$padj < 0.05] <- "UP"  
resLFC$diffexpressed[resLFC$log2FoldChange < 0.1 & resLFC$padj < 0.05] <- "DOWN"  
  
resLFC$delabel <- NA  
  
# volcano plot (note noise is removed)  
ggplot(data = resLFC, aes(x = log2FoldChange, y = log10(pvalue), col = diffexpressed, label = delabel)) +  
 geom\_point() +  
 theme\_minimal() +  
 geom\_text\_repel() +  
 scale\_color\_manual(values = c("blue","black","red")) +  
 theme(text = element\_text(size = 20))

## Warning: Removed 8674 rows containing missing values (`geom\_text\_repel()`).



## GO-Analysis

While done off-screen using Galaxy, a GO Analysis showed that 60 GO terms (0.50%) are over-represented and 7 (0.07%) under-represented and that. The Go terms are represented as such: over-represented GO terms, 50 BP, 5 CC and 5 MF and for under-represented, 5 BP, 2 CC and 0 MF.

## Kegg Pathway Analysis

When performing a Kegg pathways analysis, it was noted that around 2% of pathways were over-represented. The KEGG pathways over-represented are 01100 and 00010. From the Kegg database: 01100 corresponds to all metabolic pathways and 00010 to pathway for Glycolysis. It appears that no pathways are under-represented. You may find the pathway image in the data folder.

## Future Goals:

This project was very useful in understanding ways to filter, visualize, and understand differentially expressed genes. I feel the experience gained from using DESeq2 in the project will be invaluable, and performed a Kegg Pathway Analysis alongside the GO-Analysis was a good experience, despite them being a great source of strife. In the future, I wish to understand more about the individual genes I work with, as I felt a bit of a disconnect in matching the genes to what they actually do. For example, I seek to answer questions such as: How can these results be applied?

I also hope to perform similar analyses using UNIX or Python instead of R to gain more experience in different methods of analysis.

## Sources

<https://academic.oup.com/bioinformatics/article/32/19/3047/2196507?login=true> <https://www.youtube.com/playlist?list=PLe1-kjuYBZ05N8tWd2XVW67C4SJOJIdXD> <https://www.youtube.com/watch?v=JPwdqdo_tRg> <https://training.galaxyproject.org/training-material/topics/transcriptomics/tutorials/goenrichment/tutorial.html> <https://training.galaxyproject.org/training-material/topics/transcriptomics/tutorials/ref-based/tutorial.html#conclusion> <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3032923/> <https://www.genome.jp/kegg/>