DESeq

## Loading packages

#if installation is need, uncomment the following  
#if (!require("DESeq2")) install.packages("DESeq2"); library(DESeq2)  
#if (!require("apeglm")) install.packages("apeglm"); library(apeglm)  
#if (!require("ggplot2")) install.packages("ggplot2"); library(ggplot2)  
#if (!require("pheatmap")) install.packages("pheatmap"); library(pheatmap)  
  
library(DESeq2)

## Loading required package: S4Vectors

## Loading required package: stats4

## Loading required package: BiocGenerics

##   
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:stats':  
##   
## IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':  
##   
## anyDuplicated, append, as.data.frame, basename, cbind, colnames,  
## dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,  
## grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,  
## order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,  
## rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,  
## union, unique, unsplit, which.max, which.min

##   
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:base':  
##   
## expand.grid, I, unname

## Loading required package: IRanges

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##   
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':  
##   
## colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,  
## colCounts, colCummaxs, colCummins, colCumprods, colCumsums,  
## colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,  
## colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,  
## colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,  
## colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,  
## colWeightedMeans, colWeightedMedians, colWeightedSds,  
## colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,  
## rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,  
## rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,  
## rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,  
## rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,  
## rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,  
## rowWeightedMads, rowWeightedMeans, rowWeightedMedians,  
## rowWeightedSds, rowWeightedVars

## Loading required package: Biobase

## Welcome to Bioconductor  
##   
## Vignettes contain introductory material; view with  
## 'browseVignettes()'. To cite Bioconductor, see  
## 'citation("Biobase")', and for packages 'citation("pkgname")'.

##   
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':  
##   
## rowMedians

## The following objects are masked from 'package:matrixStats':  
##   
## anyMissing, rowMedians

library(apeglm)  
library(ggplot2)  
library(pheatmap)

## Loading in the files

#load count matrix and reference   
#tximport summerization of counts = 'intALL.csv'  
#'ref.txt' is the reference file needed by deseq  
  
data <- read.csv("intAll.csv", header=T, row.names = 1)  
info <- read.table("ref.txt", header = T, sep ="\t")

## Running DESeq

de <- DESeqDataSetFromMatrix(data, info, ~diet)

## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in  
## design formula are characters, converting to factors

## Editing DESeq output

#removes low expressed genes  
keep <- rowSums(counts(de)) >= 100  
de <- de[keep,]  
deSeqData <- DESeq(de)

## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing

###un-comment to export normalized read count  
normCounts <- counts(deSeqData, normalized = T)  
#write.csv(normCounts, "normal.allCounts.txm.csv")  
  
#p value less than .05 is diff. exp.   
result <- results(deSeqData, alpha = 0.05)  
  
# order based on p adjusted value   
resOrdered <- result[order(result$padj),]  
  
###un-comment to export ordered file   
#write.csv(resOrdered, "deSeq.order.csv")

## Editing data for plots

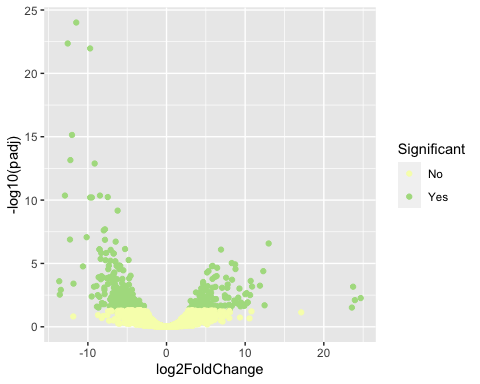
#files generated above   
normCount <- read.csv("normal.allCounts.txm.csv", row.names = 1)  
deSeqRes <- read.csv("deSeq.order.csv", row.names = 1)  
  
#if padj is <= .05, the value is significant   
deSeqRes$Significant <- ifelse(deSeqRes$padj <= 0.05, "Yes", "No")  
#taking out any na values   
deSeqRes <- na.omit(deSeqRes)  
head(deSeqRes)

## baseMean log2FoldChange lfcSE stat  
## sp|Q9JK95|PERP\_MOUSE 34527.556 -11.476800 1.0338944 -11.100553  
## sp|P15104|GLNA\_HUMAN 30268.028 -12.563096 1.1751290 -10.690823  
## sp|P62629|EF1A1\_CRIGR 382510.830 -9.713977 0.9190935 -10.569084  
## sp|Q03265|ATPA\_MOUSE 77162.053 -12.018393 1.3436407 -8.944648  
## sp|P05123|KCRM\_CANLF 99256.050 -12.224908 1.4554660 -8.399308  
## sp|Q76B49|CD63\_FELCA 1583.591 -9.136207 1.1001157 -8.304770  
## pvalue padj Significant  
## sp|Q9JK95|PERP\_MOUSE 1.246696e-28 9.717997e-25 Yes  
## sp|P15104|GLNA\_HUMAN 1.123696e-26 4.379605e-23 Yes  
## sp|P62629|EF1A1\_CRIGR 4.145233e-26 1.077070e-22 Yes  
## sp|Q03265|ATPA\_MOUSE 3.731359e-19 7.271486e-16 Yes  
## sp|P05123|KCRM\_CANLF 4.491166e-17 7.001728e-14 Yes  
## sp|Q76B49|CD63\_FELCA 1.000132e-16 1.299339e-13 Yes

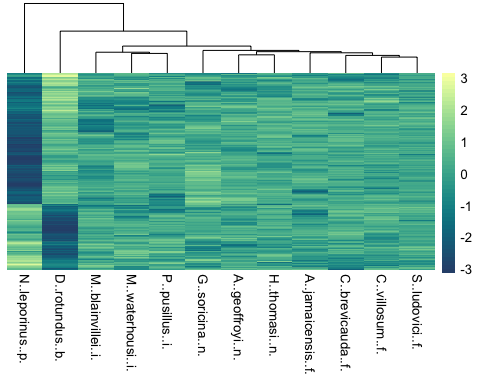
##Creating Volcano plot

#baseMean is the normalized count values, dividing by size factor, taken over all samples.. use log bc it needs to be put on a log scale   
#log2foldchage is the effect size estimate  
#ggplot(deSeqRes, aes( x = log10(baseMean), y = log2FoldChange, color = Significant)) + geom\_point()  
  
#volcano plot..

ggplot(deSeqRes, aes(x = log2FoldChange, y = -log10(padj), color = Significant)) + geom\_point() + scale\_color\_brewer(palette = "YlGn")

 ## Creating heatmap

#editing data for heat map   
signi <- subset(deSeqRes, padj <= 0.05)  
allSig <- merge(normCount, signi, by = 0)  
sigCounts <- allSig[,2:13]  
row.names(sigCounts) <- allSig$Row.names  
  
#creating heatmap  
#log2 looks at exponents instead of raw numbers, is used to normalize the data .. if there is a 0 it wont work so you have to do +1  
#scale compares expression within a col/row... finds median read count of a row/col.. doesn't look at raw numbers  
pheatmap(log2(sigCounts + 1), scale = "row", show\_rownames = F, treeheight\_row = 0, treeheight\_col = 50, color = hcl.colors(50, "BluYl"))



## Looking at the outliers from Volcano plot

#adding in column to show -log10(padj) values that are seen on the volcano plot above  
deSeqRes$neglog10 <- (-log10(deSeqRes$padj))  
##neglog10 values above 10 are observed to be outliers according to the volcano plot   
head(deSeqRes)

## baseMean log2FoldChange lfcSE stat  
## sp|Q9JK95|PERP\_MOUSE 34527.556 -11.476800 1.0338944 -11.100553  
## sp|P15104|GLNA\_HUMAN 30268.028 -12.563096 1.1751290 -10.690823  
## sp|P62629|EF1A1\_CRIGR 382510.830 -9.713977 0.9190935 -10.569084  
## sp|Q03265|ATPA\_MOUSE 77162.053 -12.018393 1.3436407 -8.944648  
## sp|P05123|KCRM\_CANLF 99256.050 -12.224908 1.4554660 -8.399308  
## sp|Q76B49|CD63\_FELCA 1583.591 -9.136207 1.1001157 -8.304770  
## pvalue padj Significant neglog10  
## sp|Q9JK95|PERP\_MOUSE 1.246696e-28 9.717997e-25 Yes 24.01242  
## sp|P15104|GLNA\_HUMAN 1.123696e-26 4.379605e-23 Yes 22.35857  
## sp|P62629|EF1A1\_CRIGR 4.145233e-26 1.077070e-22 Yes 21.96776  
## sp|Q03265|ATPA\_MOUSE 3.731359e-19 7.271486e-16 Yes 15.13838  
## sp|P05123|KCRM\_CANLF 4.491166e-17 7.001728e-14 Yes 13.15479  
## sp|Q76B49|CD63\_FELCA 1.000132e-16 1.299339e-13 Yes 12.88628

## Plotting just the outliers

signi <- subset(deSeqRes, neglog10 >= 10)  
allSig <- merge(normCount, signi, by = 0)  
sigCounts <- allSig[,2:13]  
row.names(sigCounts) <- allSig$Row.names  
pheatmap(log2(sigCounts + 1), scale = "row", show\_rownames = F,treeheight\_row = 0,color = hcl.colors(50, "YlGn"))

