STAR_DGE

John Reddy Peasari

working directory

```
setwd("D:/Spring 2020/BI-2/Labs/RNA_seq_HW/Counts_ncbi")
```

read in count matrix

```
countData <- read.csv("counts.txt", header=T, row.names=1, sep="\t")
dim(countData)
```

```
## [1] 6420    4
```

```
head(countData)
```

```
##                 SRR1066657 SRR1066658 SRR1066659 SRR1066660
## gene-YAL068C             0          2          0          0
## gene-YAL067W-A           1          4          4          0
## gene-YAL067C           105        102        246        378
## gene-YAL065C           180        184        155        223
## gene-YAL064W-B         208        231        257        242
## gene-YAL064C-A         882        841       1355       1037
```

```
nrow(countData)
```

```
## [1] 6420
```

```
par("mar")
```

```
## [1] 5.1 4.1 4.1 2.1
```

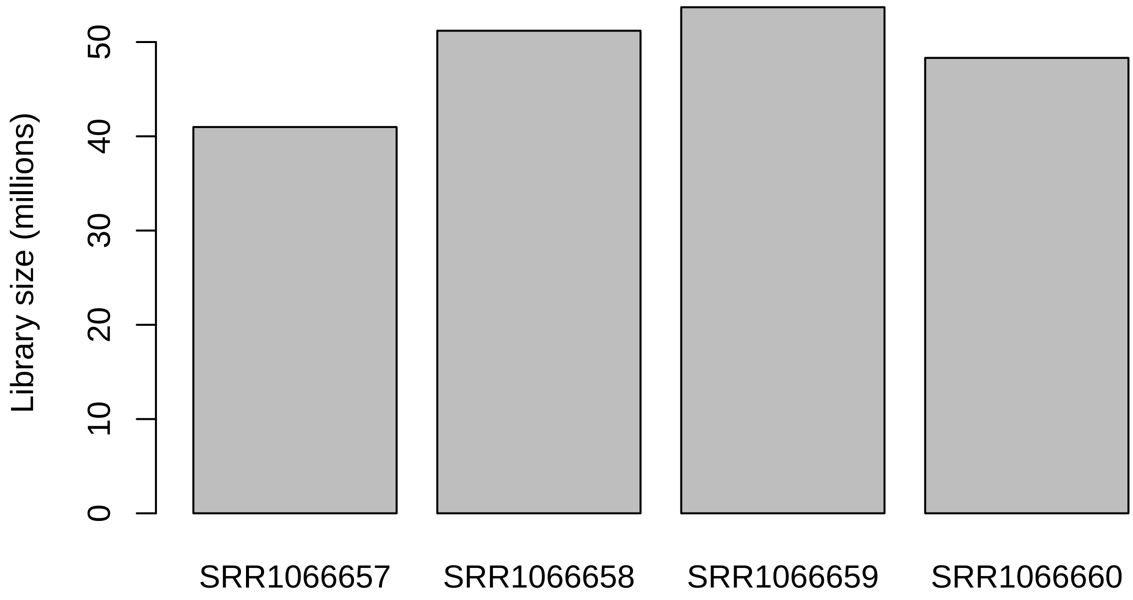
```
par(mar=c(3,3,3,3))
```

basic QC

```
barplot(colSums(countData)*1e-6,mes=colnames(countData),ylab="Library size (millions)")
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "mes" is not a graphical
## parameter
```

```
## Warning in axis(if (horiz) 2 else 1, at = at.l, labels = names.arg, lty =
## axis.lty, : "mes" is not a graphical parameter
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "mes"
## is not a graphical parameter

## Warning in axis(if (horiz) 1 else 2, cex.axis = cex.axis, ...): "mes" is not a
## graphical parameter
```



load library

create experiment labels (two conditions)

```
colData <- DataFrame(condition=factor(c("WT_NR","WT_NR","WT_CR", "WT_CR")))
colData
```

```
## DataFrame with 4 rows and 1 column
##   condition
##    <factor>
## 1     WT_NR
## 2     WT_NR
## 3     WT_CR
## 4     WT_CR
```

create DESeq input matrix

```
dds <- DESeqDataSetFromMatrix(countData, colData, formula(~ condition))
```

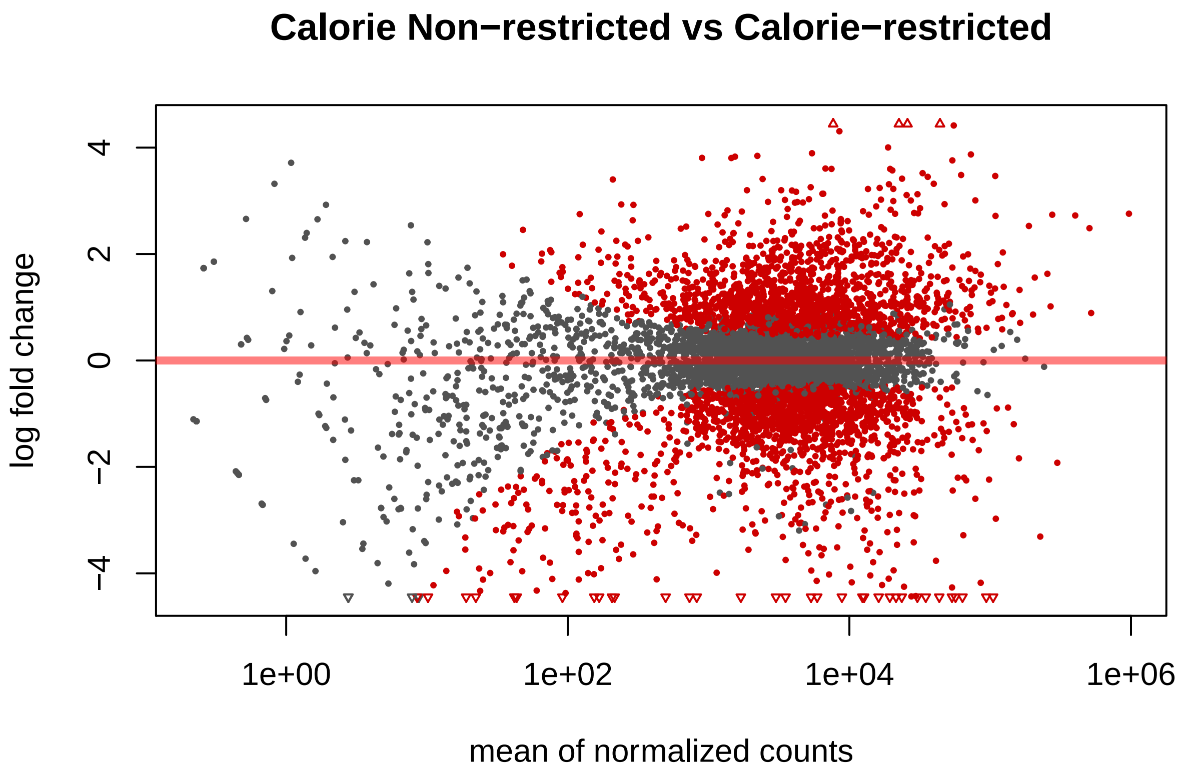
run DEseq

```
dds <- DESeq(dds)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

visualize differentially expressed genes

```
plotMA(dds,main = "Calorie Non-restricted vs Calorie-restricted")
```



get differentially expressed genes

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

```
## log2 fold change (MLE): condition WT NR vs WT CR
## Wald test p-value: condition WT NR vs WT CR
## DataFrame with 6420 rows and 6 columns
##                          baseMean    log2FoldChange             lfcSE
##                         <numeric>         <numeric>         <numeric>
## gene-YAL068C    0.51864875835297  2.65984496651804  4.90092618394776
## gene-YAL067W-A  2.22120851627164 0.616584586252061  3.58377117464893
## gene-YAL067C      199.7524216288 -1.26772349311212 0.490283961289339
## gene-YAL065C    188.307278892317 0.268443584635094 0.477090604141482
## gene-YAL064W-B  235.832474693373 0.140466055525969 0.393817651973669
## ...                          ...               ...               ...
## gene-tF(GAA)Q   41.7428427401465 -4.92842601867923  1.62283738616967
## gene-tT(UAG)Q2 0.461147229492184 -2.14927282053125  4.88991024189487
## gene-tV(UAC)Q   2.88911166941115 -1.31565252876827  2.81566901616445
## gene-tM(CAU)Q2 0.680492898317599 -2.71302317256288  4.85614406134939
## gene-Q0285      2.53137668887537 -3.04019241224947  3.40653358490928
##                              stat              pvalue                padj
##                         <numeric>           <numeric>           <numeric>
## gene-YAL068C    0.542722919441219   0.587320591346901   0.688271724501525
## gene-YAL067W-A  0.172049094711652   0.863398932415388   0.905922407289756
## gene-YAL067C    -2.58569236035845 0.00971836278383678  0.0246120464718373
## gene-YAL065C    0.562667934150903   0.573661023176157    0.67646372305216
## gene-YAL064W-B   0.35667790618832   0.721332947372391    0.80083196586801
## ...                           ...                 ...                 ...
## gene-tF(GAA)Q   -3.03691920132036  0.0023900951304571 0.00752844150482642
## gene-tT(UAG)Q2 -0.439532161984714   0.660275983115513   0.751309360649893
## gene-tV(UAC)Q  -0.467261074087632   0.640313102563653   0.734949840692828
## gene-tM(CAU)Q2 -0.558678477880454   0.576381170166019   0.678266439028159
## gene-Q0285     -0.892459251162919   0.372146824742966   0.492705831792808
```

```r
summary(res)
```

```
##
## out of 6306 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 1564, 25%
## LFC < 0 (down)     : 1605, 25%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 0)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

order by BH adjusted p-value

```r
resOrdered <- res[order(res$padj),]
```

top of ordered matrix

```r
head(resOrdered)
```

```
## log2 fold change (MLE): condition WT NR vs WT CR
## Wald test p-value: condition WT NR vs WT CR
## DataFrame with 6 rows and 6 columns
##                       baseMean    log2FoldChange             lfcSE
##                      <numeric>         <numeric>         <numeric>
## gene-YCR010C 93813.9425252378 -7.60147198535194 0.268852144003086
## gene-YDR345C 44004.0888488117  6.10042065916196  0.22796313318669
## gene-YMR303C 63561.3211090224 -6.70868252315592 0.252271180718402
## gene-YOL154W 25862.4816457531  5.53560915245836 0.219403641751068
## gene-YIL057C 12719.3656552844 -6.52771400521533 0.260266813283609
## gene-YKL217W  104882.17650244 -6.52839115570243 0.261812798193852
##                           stat                pvalue                  padj
##                      <numeric>             <numeric>             <numeric>
## gene-YCR010C -28.2738008786893 7.25792736163344e-176 4.57684899424605e-172
## gene-YDR345C  26.7605580511391 9.30543919971862e-158 2.93400497967128e-154
## gene-YMR303C -26.5931387963197 8.14908229345056e-156 1.71293709808331e-152
## gene-YOL154W  25.2302519150479 1.86583236731327e-140 2.94148472706938e-137
## gene-YIL057C  -25.080854231316 8.04639270686945e-139 1.01481104819038e-135
## gene-YKL217W -24.9353400625918 3.07986695414747e-137 3.23694016880899e-134
```

how many differentially expressed genes ? FDR=10%, |fold-change|>2 (up and down) get differentially expressed gene matrix

```
sig <- resOrdered[!is.na(resOrdered$padj) &
                    resOrdered$padj<0.10 &
                    abs(resOrdered$log2FoldChange)>=1,]
```

top 50 of the differentially expressed genes

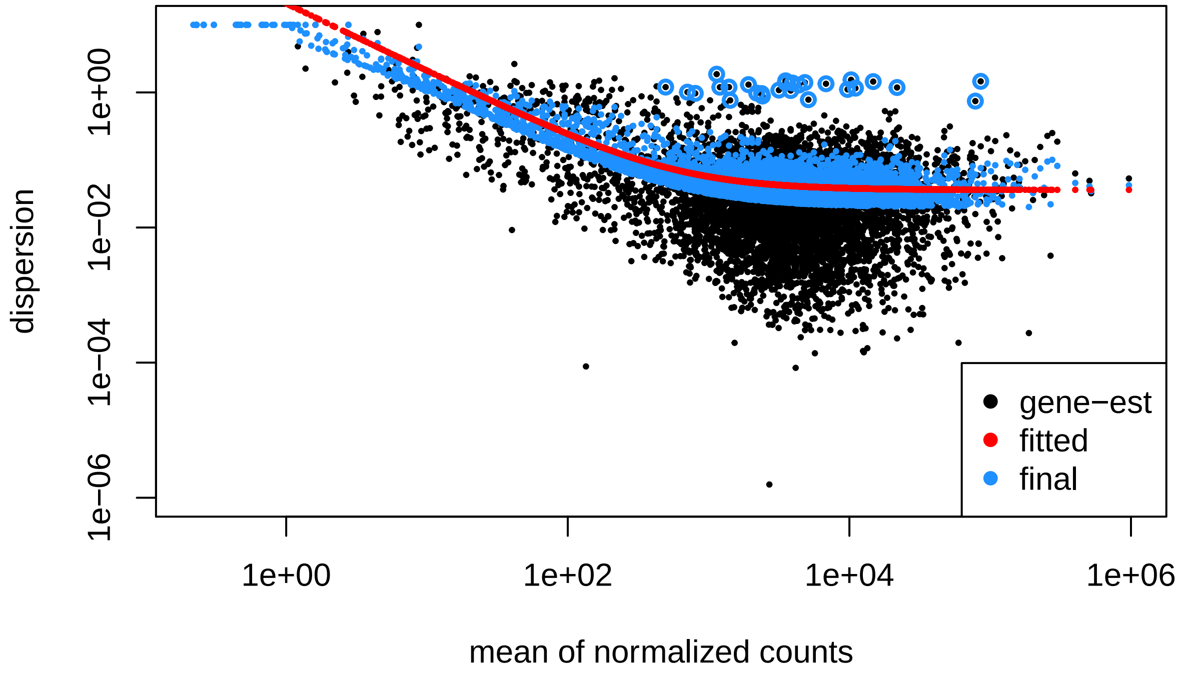
```
data <- data.frame(sig)
write.csv(data,"gene_list_ncbi.csv") ## Writing DE genes to a csv file
genes <- read.csv("gene_list_ncbi.csv") ## Reading csv file
head(genes$X,n=50)
```

```
##  [1] gene-YCR010C   gene-YDR345C   gene-YMR303C   gene-YOL154W   gene-YIL057C  
##  [6] gene-YKL217W   gene-YER024W   gene-YPR001W   gene-YHR137W   gene-YAL054C  
## [11] gene-YDR040C   gene-YLR327C   gene-YOR100C   gene-YER179W   gene-YOR348C  
## [16] gene-YML054C   gene-YGL029W   gene-YKR097W   gene-YGR067C   gene-YKL172W  
## [21] gene-YGL205W   gene-YDR256C   gene-YLR377C   gene-YKL082C   gene-YPR002W  
## [26] gene-YNL036W   gene-YER065C   gene-YHL028W   gene-YOR310C   gene-YPL095C  
## [31] gene-YMR319C   gene-YPR006C   gene-YGR236C   gene-YMR206W   gene-YPL113C  
## [36] gene-YKR080W   gene-YLR223C   gene-YBR054W   gene-YBR092C   gene-YPL135W  
## [41] gene-YGL256W   gene-YDL215C   gene-YMR107W   gene-YOL052C-A gene-YNL308C  
## [46] gene-YMR120C   gene-YGR043C   gene-YOR051C   gene-YNL065W   gene-YDL214C  
## 1813 Levels: gene-IRT1 gene-LSR1 gene-Q0020 gene-RDN5-1 gene-RME2 ... gene-YPR199C
```

```
#head(genes$Gene.Name, n = 50) ### Getting top 50 DE genes
```

Dispersion plot

```
plotDispEsts( dds, ylim = c(1e-6, 1e1) )
```

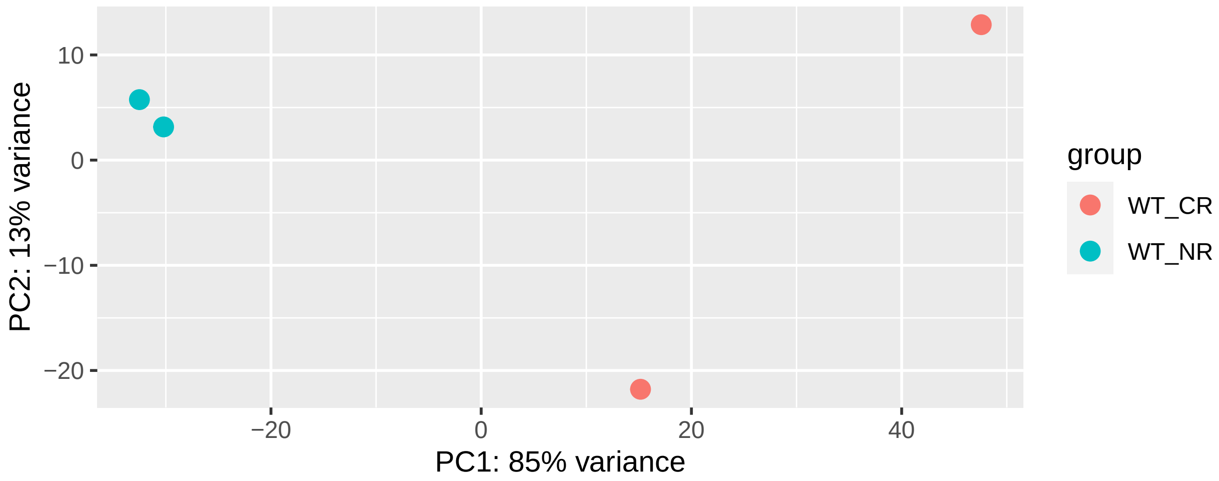


```
rld <- rlog( dds )
rld
```

```
## class: DESeqTransform
## dim: 6420 4
## metadata(1): version
## assays(1): ''
## rownames(6420): gene-YAL068C gene-YAL067W-A ... gene-tM(CAU)Q2
##   gene-Q0285
## rowData names(23): baseMean baseVar ... dispFit rlogIntercept
## colnames(4): SRR1066657 SRR1066658 SRR1066659 SRR1066660
## colData names(2): condition sizeFactor
```

the call to DESeqTransform() is needed to trigger our plotPCA method

```
library("RColorBrewer")
hmcol <- colorRampPalette(brewer.pal(9, "GnBu"))(100) ## hmcol <- heat.colors

se <- SummarizedExperiment(log2(counts(dds, normalized=TRUE) + 1),
                           colData=colData(dds))
plotPCA( DESeqTransform( se ) )
```


PC2: 13% variance
10
0
−10
−20
−20
0
20
40
PC1: 85% variance
group
WT_CR
WT_NR