Differential Expression Analysis

First we will load the necessary libraries.

```
library(tidyverse)
library(DESeq2)
library(magrittr)
library(edgeR)
```

The data has been extracted from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE226134 from the GSE226134_CK_10___norm.xlsx file. Lauren Mock selected for pre-treatment samples and performed data quality control.

```
intCountData <- read.csv("data/input/Integer_mRNA_counts.csv",row.names=1)
property <- read.csv("data/input/Patient_properties.csv")
```

DESeq

```
dds <- DESeqDataSetFromMatrix(countData = intCountData,
                              colData = property,
                              design = ~METASTATIC)
```

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

DESeq recommends that row counts are filtered to remove rows with very few reads, especially rows with less than 10 reads. Here it appears that we end up keeping all rows.

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

Next we will run DESeq to get the differentially expressed genes.

```
dds$METASTATIC <- relevel(dds$METASTATIC, ref = "False")
dds <- DESeq(dds)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.
```

```
final dispersion estimates
```

```
fitting model and testing
```

```
-- replacing outliers and refitting for 116 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)
```

```
estimating dispersions
```

```
fitting model and testing
```

```
adj_pval_threshold <- 0.05
res <- results(dds, alpha = adj_pval_threshold)
summary(res)
```

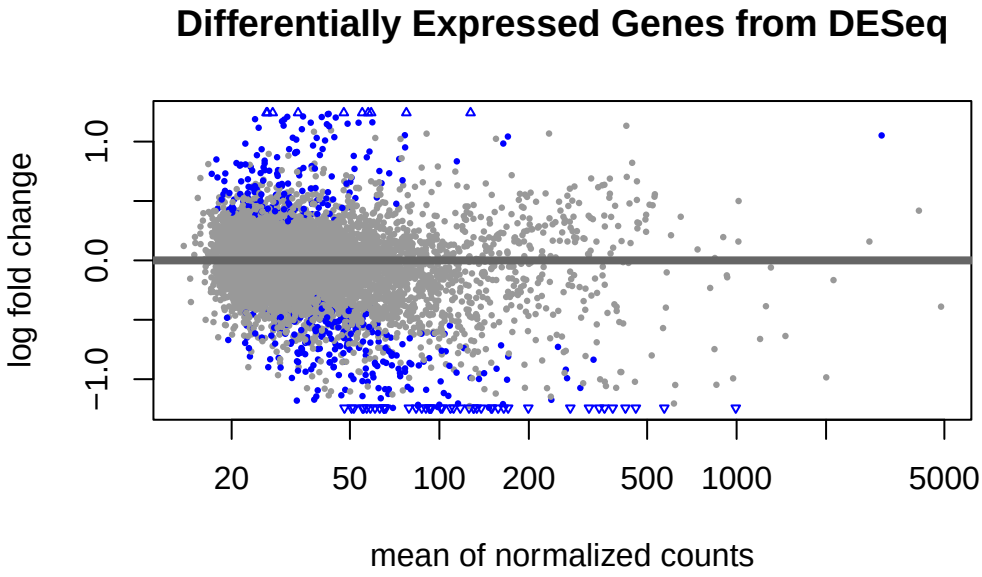
```
out of 9223 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 195, 2.1%
LFC < 0 (down)     : 281, 3%
outliers [1]       : 0, 0%
low counts [2]     : 0, 0%
(mean count < 14)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

```
resultsNames(dds)
```

```
[1] "Intercept"                 "METASTATIC_True_vs_False"
```

```
# get top 5 most differentially expressed genes

DESeq2::plotMA(res, main="Differentially Expressed Genes from DESeq")
```




```
resOrdered <- res[order(res$padj),]
resSig <- subset(resOrdered, padj < 0.05)
resSig <- subset(resSig, abs(log2FoldChange) > 1)
```

```
nrow(resSig) #113 genes
```

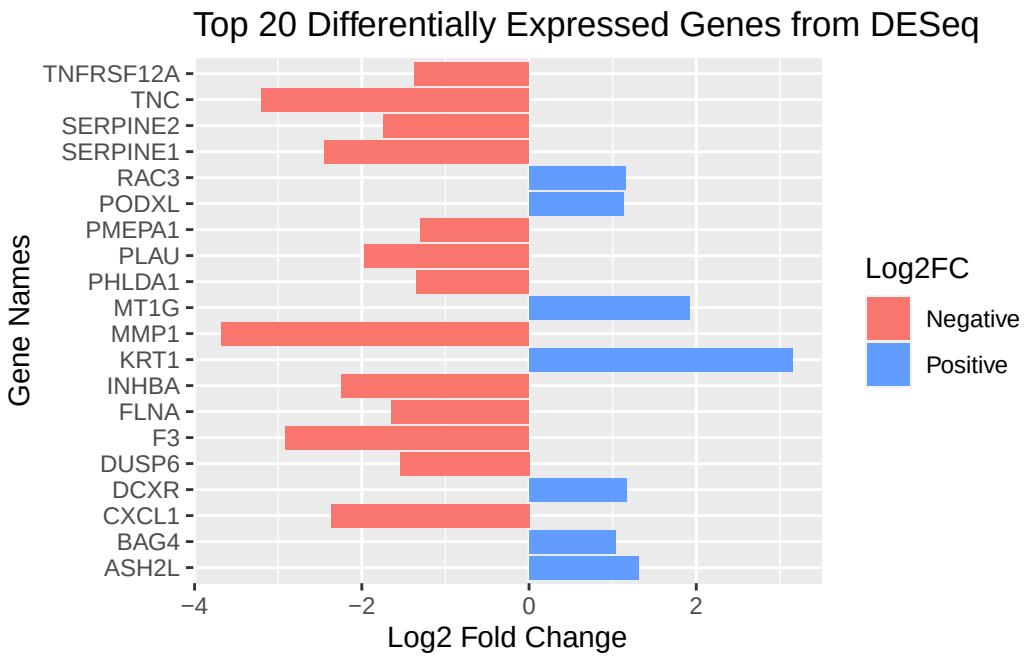
```
[1] 113
```

```
gene_names <- rownames(resSig)

deseqResultsDF <- as.data.frame(resSig@listData)
deseqResultsDF <- cbind(gene_names, deseqResultsDF)

deseqResultsDF20 <- deseqResultsDF[1:20,]

ggplot(data=deseqResultsDF20, aes(x=gene_names, y=log2FoldChange, fill=log2FoldChange > 0)
  geom_col() +
  scale_fill_manual(values=c("#F8766D","#619CFF" ), labels=c("Negative", "Positive"), name
  coord_flip() +
  scale_x_discrete(guide = guide_axis(n.dodge=1)) +
  labs(x = "Gene Names", y = "Log2 Fold Change", title = "Top 20 Differentially Expressed
```




```
#uncomment the following lines to get a printed list to input for ShinyGO
# for (gene in gene_names) {
#   cat(gene, "\n")
```

```
# } #running these genes through ShinyGO shows cancer and metabolism pathways http://bioin
```

We will save the differential expression from DESeq for the downstream analyses.

```
deseq_normalized_reads <- rbind(t(property),counts(dds,normalized = T))
colnames(deseq_normalized_reads) <- as.character(unlist(deseq_normalized_reads[1, ]))
deseq_normalized_reads <- deseq_normalized_reads[-1, ]
write.table(deseq_normalized_reads, file="data/output/DEseq_Normalized.txt",sep="\t",quote


write.csv(resSig,file="data/output/deseq_diff_exp_results.csv")
```

EdgeR

Reference: https://web.stanford.edu/class/bios221/labs/rnaseq/lab_4_rnaseq.html

First we will prepare the data and calculate the dispersion so we will next be able to find the differential expression

```
d <- DGEList(counts=intCountData,group=property$METASTATIC)
dim(d)
```

```
[1] 9223   59
```

```
#head(d$counts)
apply(d$counts, 2, sum)
```

```
YTMA496_1_13...10...Segment.3 YTMA496_1_13...15...Segment.3
                       910619                         49480
YTMA496_1_13...16...Segment.3 YTMA496_1_13...19...Segment.3
                       131443                        348687
YTMA496_1_13...20...Segment.3 YTMA496_1_13...21...Segment.3
                       915774                       1045122
YTMA496_1_13...22...Segment.3 YTMA496_1_13...23...Segment.3
                       649427                        378320
YTMA496_1_13...28...Segment.3 YTMA496_1_13...30...Segment.3
                       380713                       2014498
YTMA496_1_13...32...Segment.3 YTMA496_1_13...34...Segment.3
                       823838                        802316
```

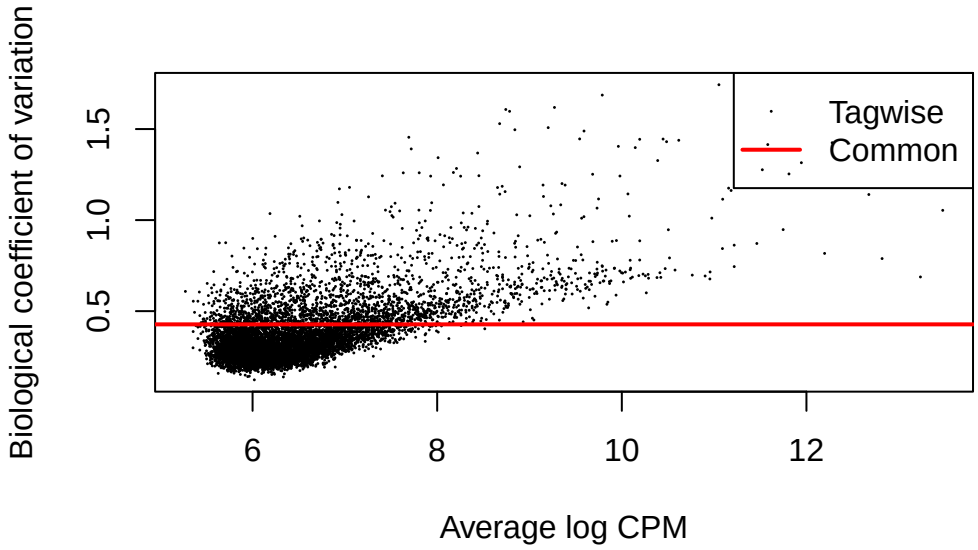
```
YTMA496_1_13...35...Segment.3 YTMA496_1_13...37...Segment.3
                       238557                         883929
 YTMA496_1_13...4...Segment.3 YTMA496_1_13...40...Segment.3
                       514514                         496125
YTMA496_1_13...41...Segment.3 YTMA496_1_13...43...Segment.3
                       361452                         314568
YTMA496_1_13...46...Segment.3 YTMA496_1_13...49...Segment.3
                       442702                        1115658
 YTMA496_1_13...5...Segment.3 YTMA496_1_13...50...Segment.3
                       369882                         156975
YTMA496_1_13...55...Segment.3 YTMA496_1_13...56...Segment.3
                       488535                        1658825
YTMA496_1_13...59...Segment.3  YTMA496_1_13...6...Segment.3
                      2145876                         485456
YTMA496_1_13...64...Segment.3 YTMA496_1_13...66...Segment.3
                       291752                        2437146
 YTMA496_1_13...9...Segment.3 YTMA496_2_13...10...Segment.3
                       381860                         524198
YTMA496_2_13...12...Segment.3 YTMA496_2_13...15...Segment.3
                        39357                          42942
YTMA496_2_13...16...Segment.3 YTMA496_2_13...19...Segment.3
                       970369                          86043
YTMA496_2_13...21...Segment.3 YTMA496_2_13...22...Segment.3
                       883825                         254777
YTMA496_2_13...23...Segment.3 YTMA496_2_13...28...Segment.3
                      1124356                         122495
YTMA496_2_13...30...Segment.3 YTMA496_2_13...34...Segment.3
                       309227                         567186
YTMA496_2_13...35...Segment.3 YTMA496_2_13...37...Segment.3
                      2001258                         517700
 YTMA496_2_13...4...Segment.3 YTMA496_2_13...40...Segment.3
                       323444                          57887
YTMA496_2_13...41...Segment.3 YTMA496_2_13...43...Segment.3
                       528169                        1187802
YTMA496_2_13...47...Segment.3 YTMA496_2_13...49...Segment.3
                      1153804                         211023
YTMA496_2_13...50...Segment.3 YTMA496_2_13...53...Segment.3
                        64074                         108979
YTMA496_2_13...54...Segment.3 YTMA496_2_13...55...Segment.3
                       528054                         500160
YTMA496_2_13...56...Segment.3  YTMA496_2_13...6...Segment.3
                       802961                        2543986
YTMA496_2_13...61...Segment.3 YTMA496_2_13...63...Segment.3
```

```
                        474835                          303885
YTMA496_2_13...64...Segment.3 YTMA496_2_13...66...Segment.3
                        152894                         1453165
YTMA496_2_13...69...Segment.3
                       1118159
```

```
#filtering steps for DESeq
keep <- rowSums(cpm(d)>100) >= 2
d <- d[keep,]
dim(d) #cuts down about 600 genes
```

```
[1] 8680   59
```

```
d$samples$lib.size <- colSums(d$counts)
d <- calcNormFactors(d)
d1 <- estimateCommonDisp(d)
d1 <- estimateTagwiseDisp(d1)
plotBCV(d1)
```



We will now use our information from the dispersion calculation to check for differential expression and then compare to the results from DESeq.

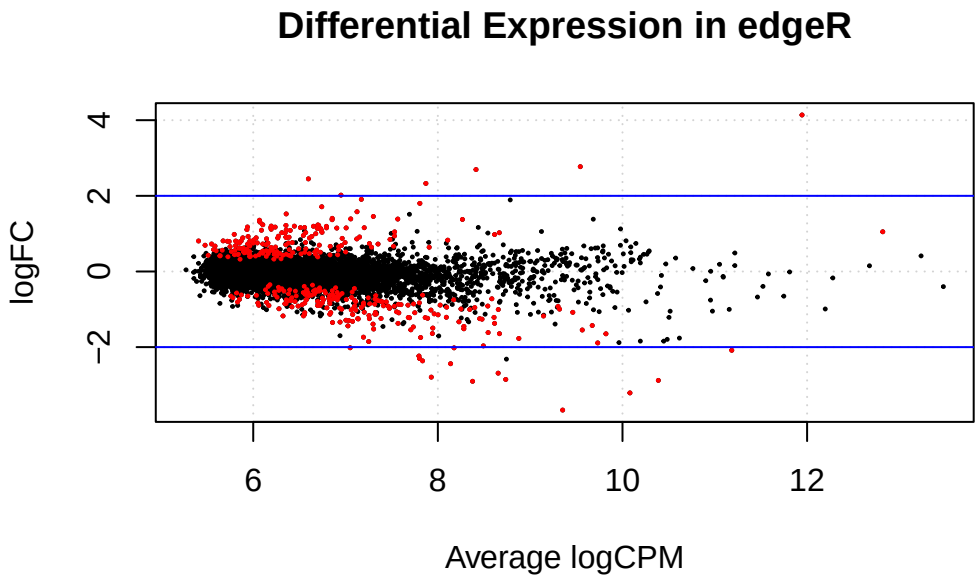
```
et12 <- exactTest(d1, pair=c(1,2))
topTags(et12, n=10)
```

```
Comparison of groups:  True-False
           logFC    logCPM       PValue          FDR
PITX1  4.1377711 11.945164 1.817350e-15 1.577460e-11
MT1G   1.9075098  7.171275 5.273978e-13 1.806617e-09
ANK1   2.6948864  8.413772 6.244069e-13 1.806617e-09
GSTA1  2.4484216  6.597395 4.285723e-12 9.300019e-09
DCXR   1.1614314  6.191021 4.497128e-11 7.807015e-08
ASH2L  1.3051355  6.072951 1.168861e-09 1.690953e-06
TNC   -3.2101590 10.080863 1.991779e-09 2.438533e-06
BAG4   1.0346380  6.745474 2.247496e-09 2.438533e-06
LUM    2.3276325  7.870496 2.841152e-09 2.740133e-06
FGFR1  0.9251594  5.944759 4.159412e-09 3.610369e-06
```

```r
de1 <- decideTestsDGE(et12, adjust.method="BH", p.value=0.05)
summary(de1)
```

```
       True-False
Down          211
NotSig       8296
Up            173
```

```r
de1tags12 <- rownames(d1)[as.logical(de1)]
plotSmear(et12, de.tags=de1tags12, main="Differential Expression in edgeR")
abline(h = c(-2, 2), col = "blue")
```



```
tags <- topTags(et12, n=Inf)
top_genes <- rownames(tags$table)[tags$table$FDR < 0.05 & abs(tags$table$logFC) > 1]

# for (gene_name in top_genes) {
#   cat(gene_name, "\n")
# }
```

Compare DESeq and EdgeR

```
sum(top_genes %in% rownames(resSig)) #106 of 113 genes match
```

```
[1] 106
```

```
sum(rownames(resSig) %in% top_genes) #106; serves as a check
```

```
[1] 106
```

```
gene <- top_genes[top_genes %in% rownames(resSig)]
#the output of the following lines is very long so it will be omitted from our rendered do
#to get the list of genes for ShinyGO please uncomment the following lines:
# for (gene_name in gene) {
#   cat(gene_name, "\n")
# }
```

We can see that there are 106 genes in common between the results from DESeq and from EdgeR. When we plug in the overlapping genes into ShinyGO, we see pathways enriched for receptor interactions, cancer, adhesion, and signaling pathways, which make sense given the biological basis of metastasis.

We will use these 106 genes as the differential expression genes for downstream steps in the process. We will use the p-value information from DESeq.

```
res_match <- subset(resSig, rownames(resSig) %in% top_genes)

res_match_df <- cbind(gene,as.data.frame(res_match@listData))
write.csv(res_match_df,file="data/output/deseq_edger_overlap_diff_exp_results.csv")
```