Class 12: RNA Seq analysis

Serena Quezada (PID:A18556865)

Table of contents

Background

Today we will analyze some RNASeq data from Hime et al. on the effects of a common steroid (dexamethasone) on airway smooth muscle cells (ASM cells).

Our starting point is the "counts" data and metadata the count value for each gene in their different experiments (i.e. cell lines with or without the drug).

Data Import

```
# Complete the missing code
counts <- read.csv("airway_scaledcounts.csv", row.names = 1)
metadata <- read.csv("airway_metadata.csv")
```

```
head(counts)
```

	SRR1039508	SRR1039509	SRR1039512	SRR1039513	SRR1039516
ENSG00000000003	723	486	904	445	1170
ENSG00000000005	0	0	0	0	0
ENSG00000000419	467	523	616	371	582
ENSG00000000457	347	258	364	237	318

```
ENSG00000000460          96          81          73          66         118
ENSG00000000938           0           0           1           0           2
                SRR1039517 SRR1039520 SRR1039521
ENSG00000000003       1097         806         604
ENSG00000000005          0           0           0
ENSG00000000419        781         417         509
ENSG00000000457        447         330         324
ENSG00000000460         94         102          74
ENSG00000000938          0           0           0
```

Q1. How many genes are in this dataset?

```
nrow(counts)
```

```
[1] 38694
```

Q. How many different experiments (columns in counts or rows in metadata are there?)

This is the amount of experiments

```
nrow(metadata)
```

```
[1] 8
```

```
head(metadata)
```

```
          id     dex celltype     geo_id
1 SRR1039508 control   N61311 GSM1275862
2 SRR1039509 treated   N61311 GSM1275863
3 SRR1039512 control  N052611 GSM1275866
4 SRR1039513 treated  N052611 GSM1275867
5 SRR1039516 control  N080611 GSM1275870
6 SRR1039517 treated  N080611 GSM1275871
```

Q2. How many 'control' cell lines do we have?

```
sum(metadata$dex == "control")
```

```
[1] 4
```

Toy differential gene expression

To start analysis let's calculate the mean counts for all genes in the "control" experiments.

1. Extract all "control" columns from the `counts` object
2. Calculate the mean for all rows (genes) of these "control" columns

3-4. Do the same for the "treated" 5. Compare the `control.mean` and `treated.mean` values.

> Q3. How would you make the above code in either approach more robust? Is there a function that could help here?

Below is how I would make the code more robust, considering this a better function to help find the mean values of both control and treated values.

This extracts all the "controls"

```
control.inds <- metadata$dex == "control"
control.counts <- counts[ , control.inds]
```

This is calculating the means for these `control` means

```
control.means <- rowMeans(control.counts)
```

> Q4. Follow the same procedure for the treated samples (i.e. calculate the mean per gene across drug treated samples and assign to a labeled vector called treated.mean)

Now we are calculating the treated columns

```
treated.inds <- metadata$dex == "treated"
treated.counts <- counts[ , treated.inds]
```

```
treated.means <- rowMeans(treated.counts)
```

Now we are comparing the control and treated mean values (mean value per gene)

```
meancounts <- data.frame(control.means, treated.means)
head(meancounts)
```

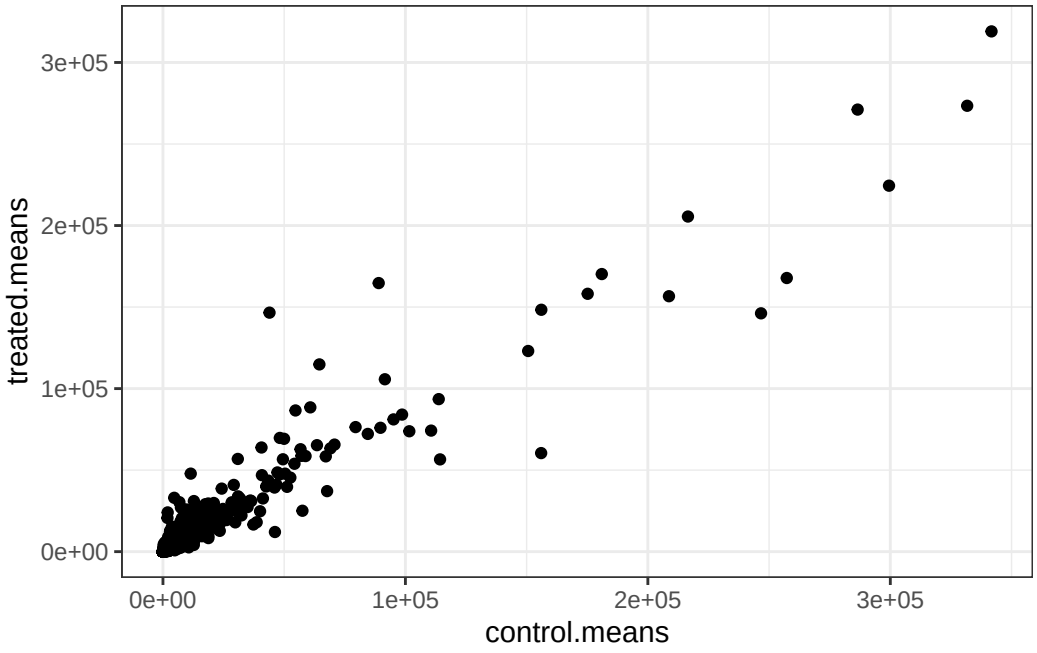
```
                control.means treated.means
ENSG00000000003        900.75        658.00
ENSG00000000005          0.00          0.00
ENSG00000000419        520.50        546.00
ENSG00000000457        339.75        316.50
ENSG00000000460         97.25         78.75
ENSG00000000938          0.75          0.00
```

Now make a plot for our values

> Q5 (a). Create a scatter plot showing the mean of the treated samples against the mean of the control samples.
>
> Q5 (b). You could also use the ggplot2 package to make this figure producing the plot below. What geom_?() function would you use for this plot?

You would use the geom_point() to produce a plot

```
library(ggplot2)
ggplot(meancounts,
       aes(x = control.means, y = treated.means)) +
       geom_point() + theme_bw()
```



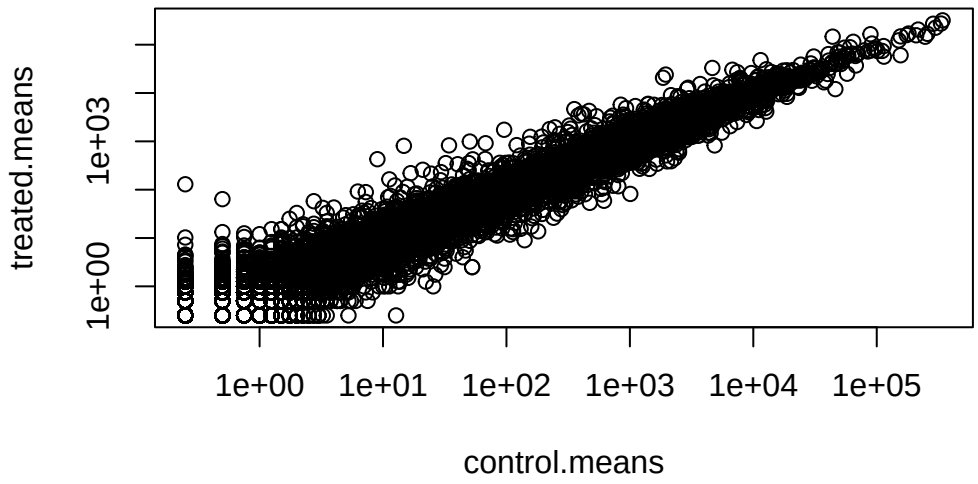
Q6. Try plotting both axes on a log scale. What is the argument to plot() that allows you to do this?

Make this a log plot, by doing this we have more plots and a direct dark linear line appears

```
plot(meancounts, log="xy")
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15032 x values <= 0 omitted
from logarithmic plot
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15281 y values <= 0 omitted
from logarithmic plot
```



We often talk metrics like "log2 fold-change"

```
# control/treated
log2(20/10)
```

```
[1] 1
```

```
log2(40/10)
```

```
[1] 2
```

```
log2(10/40)
```

```
[1] -2
```

Let's calculate the log2 fold for our treated over control mean counts We use the log2 fold to help us identify and interpret differently expressed genes between control and treated genes

```
meancounts$log2fc <-
log2(meancounts$treated.means /
       meancounts$control.means)
```

```
head(meancounts)
```

```
                control.means treated.means      log2fc
ENSG00000000003        900.75        658.00 -0.45303916
ENSG00000000005          0.00          0.00         NaN
ENSG00000000419        520.50        546.00  0.06900279
ENSG00000000457        339.75        316.50 -0.10226805
ENSG00000000460         97.25         78.75 -0.30441833
ENSG00000000938          0.75          0.00        -Inf
```

A common "rule of thumb" is a log2 fold change cutoff of +2 and -2 to call genes "Up regulated" or "Down regulated".

> Q8. Can you determine how many up regulated genes we have at the greater than 2 fc level?

```
sum(meancounts$log2fc > +2, na.rm = T)
```

```
[1] 1846
```

```
#na.rm=T means that it won't count the NA's
```

> Q9. Can you determine how many down regulated genes we have at the greater than 2 fc level?

```
sum(meancounts$log2fc < -2, na.rm = T)
```

```
[1] 2212
```

Q10. Do you trust these results? Why or why not?

When using the log2fold we are missing statistical significance. We don't know if the control and treated are caused by random error * write more on what we're missing *

##DESeq2 Analysis

Let's do this analysis properly and keep our inner stats nerd happy - i.e. are the differences we see between drug and no drug significant given the replicate experiments

```
library(DESeq2)
```

For DESeq analysis we need 3 things

- count values (`contData`)
- metadata telling us about the columns in `countData` (`colData`)
- design of the experiment (i.e. what do you want to compare)

Our analysis function form DESeq2 will setup the input required for analysis by storing all these 3 things together

```
dds <- DESeqDataSetFromMatrix(countData = counts,
                              colData = metadata,
                              design = ~dex)
```

```
converting counts to integer mode
```

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

The main function in DESeq2 that runs the analysis is called `DESeq()`

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

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates

mean-dispersion relationship

final dispersion estimates

fitting model and testing
```

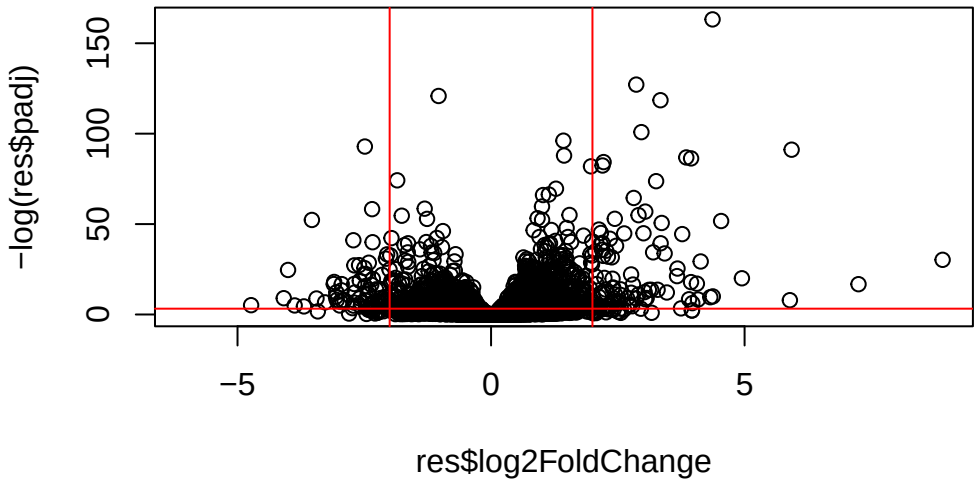
```
res <- results(dds)
res
```

```
log2 fold change (MLE): dex treated vs control
Wald test p-value: dex treated vs control
DataFrame with 38694 rows and 6 columns
                 baseMean log2FoldChange     lfcSE      stat    pvalue
                <numeric>      <numeric> <numeric> <numeric> <numeric>
ENSG00000000003  747.1942      -0.350703  0.168242 -2.084514 0.0371134
ENSG00000000005    0.0000             NA        NA        NA        NA
ENSG00000000419  520.1342       0.206107  0.101042  2.039828 0.0413675
ENSG00000000457  322.6648       0.024527  0.145134  0.168996 0.8658000
ENSG00000000460   87.6826      -0.147143  0.256995 -0.572550 0.5669497
...                    ...            ...       ...       ...       ...
ENSG00000283115  0.000000             NA        NA        NA        NA
ENSG00000283116  0.000000             NA        NA        NA        NA
ENSG00000283119  0.000000             NA        NA        NA        NA
ENSG00000283120  0.974916       -0.66825   1.69441 -0.394385  0.693297
ENSG00000283123  0.000000             NA        NA        NA        NA
                     padj
                <numeric>
ENSG00000000003  0.163017
ENSG00000000005        NA
ENSG00000000419  0.175937
ENSG00000000457  0.961682
ENSG00000000460  0.815805
...                   ...
ENSG00000283115        NA
ENSG00000283116        NA
ENSG00000283119        NA
ENSG00000283120        NA
ENSG00000283123        NA
```

Volcano plot

This is a common summary result figure from these types of experiments and plot the log2 fold-change vs the adjusted p-value.

```
plot(res$log2FoldChange, -log(res$padj))
abline(v = c(-2,2), col = "red")   #abline provides vertical cut-off lines
abline(h = -log(0.04), col = "red") # provides cut-off horizontal lines
```



Save our results

```
write.csv(res, file = "my_results.csv")
```