Homework Assignment 7
Total Points: 138
(Due Thursday November 16, 2023 at 5PM)

Please email your answer (compiled pdf file from R markdown) and R code to Cenxiao (CENXIAO@email.sc .edu).

This is in continuation of Question 2 in Homework Assignment 6. In this homework, you will need the following data files.
Data files:
sampleinfo.txt
GSE60450_Lactation-GenewiseCounts.txt
mouse_c2_v5.rdata
mouse_H_v5.rdata

These files Data files available from: https://figshare.com/s/1d788fd384d33e913a2a You need to download these files and place them in your `/data` directory. The following Bioconductor packages are used for this homework. Packages used:

- limma,
- edgeR,
- gplots,
- org.Mm.eg.db,
- RColorBrewer,
- Glimma

In this homework, we will go through the typical RNA-seq data analysis procedure.

Overview

- Reading in table of counts (Homework 6 Q2)
- Filtering lowly expressed genes (Homework 6 Q2)
- Quality control
- Normalisation for composition bias

Quality control

Now that we have got rid of the lowly expressed genes and have our counts stored in a `DGEList` object, we can look at a few different plots to check that the data is good quality, and that the samples are as we would expect.

Library sizes and distribution plots

Count data is not normally distributed, so if we want to examine the distributions of the raw counts we need to log the counts. Next we'll use box plots to check the distribution of the read counts on the log2 scale. We can use the `cpm` function to get log2 counts per million, which are corrected for the different library sizes. The `cpm` function also adds a small offset to avoid taking log of zero.

1. Load the DGEList RData object stored in Q8 in Homework 6. (2 points)
2. Make a barplot of library size for each sample. (5 points)
3. Use the 'cpm' function to transfer the raw count in $\log_2$ scale. Make a boxplot of the log-scaled raw count to examine overall the density distributions of raw log-intensities. Do any samples appear to be different compared to the others? (10 points)

Multidimensional scaling plots

By far, one of the most important plots we make when we analyse RNA-Seq data are MDSplots. An MDSplot is a visualisation of a principle components analysis, which determines the greatest sources of variation in the data. A principle components analysis is an example of an unsupervised analysis, where we don't need to specify the groups. If your experiment is well controlled and has worked well, what we hope to see is that the greatest sources of variation in the data are the treatments/groups we are interested in. It is also an incredibly useful tool for quality control and checking for outliers. We can use the `plotMDS` function to create the MDS plot.

4. Create MDS plots using the 'plotMDS' function. Make two MDS plots, color the samples according cell types and status. In addition, instead of sample names, use points ('pch=16') to label the samples. (5 points)
5. Is there something strange going on with the samples? Identify the two samples that don't appear to be in the right place. (10 points)
6. Read in the corrected sample information (SampleInfo_Corrected.txt) and repeat Step 4. (5 points)

Another alternative is to generate an interactive MDS plot using the *Glimma* package. This allows the user to interactively explore the different dimensions.

7. Use the 'glMDSPlot' function in the *Glimma* package to generate an interactive MDS plot. (3 points)
8. Based on the interactive MDS plot, what is the greatest source of variation in the data (i.e. what does dimension 1 represent)? What is the second greatest source of variation in the data? (8 points)

Hierarchical clustering with heatmaps

An alternative to `plotMDS` for examining relationships between samples is using hierarchical clustering. Heatmaps are a nice visualisation to examine hierarchical clustering of your samples. We can do this using the `heatmap.2` function from the *gplots* package. In this example `heatmap.2` calculates a matrix of euclidean distances from the logCPM for the 500 most variable genes. (Note this has more complicated code than plotting principle components using `plotMDS`.)

The *RColorBrewer* package has nicer colour schemes, accessed using the `brewer.pal` function. "RdYlBu" is a common choice, and "Spectral" is also nice.

9. Select the data for the 500 most variable genes and plot the heatmap use the logCPM count matrix created in 9 using RdY1Bu color scheme. Color the sample using cell types information. (Hint: code below) (15 points)

```
## Get some nicer colours
mypalette <- brewer.pal(11,"RdYlBu")
## http://colorbrewer2.org/#type=sequential&scheme=BuGn&n=3
morecols <- colorRampPalette(mypalette)
# Set up colour vector for celltype variable
col.cell <- c("purple","orange")[sampleinfo$CellType]
```

Normalisation for composition bias

The trimmed mean of M-values normalization method (TMM) is performed to eliminate composition biases between libraries [@robinson2010tmm]. This generates a set of normalization factors, where the product of these factors and the library sizes defines the effective library size. The `calcNormFactors` function in `edgeR` calculates the normalization factors between libraries. TMM normalization (and most scaling normalisation methods) scale relative to one sample.

10. Use the 'calcNormFactors' in 'edgeR' to calculate the normalization factors between libraries. TMM normalisation (and most scaling normalisation methods) scale relative to one sample. (5 points) Report the normalization factors for all samples. Which sample has the smallest normalization factors.

11. Pick any two samples, and use plotMD to examine the mean-difference plot using logcount matrix before and after normalization (side by side). Do you observe any difference in the mean-difference plot? (10 points)

Differetial expression analysis for RNAseq count data

Now that we are happy that we have normalised the data and that the quality looks good, we can continue to testing for differentially expressed genes. There are a number of packages to analyse RNA-Seq data. Most people use DESEQ2 or edgeR. We will use edgeR for the rest of this practice.

Create the design matrix

12. Create a design matrix for the analysis for the two variables: status and cell type. Consider a model with main effects, and interaction of these two factors. (10 points)

Data exploration

13. An MDS plot shows distances, in terms of biological coefficient of variation (BCV), between samples. Create a MDS plot using the data from all samples. Label the samples according to their cell type and status. [Hint:plotMDS] (5 points)

14. What do you think of the quality of the data? Can you anticipate if the interaction term will be important? (10 points)

Estimating the dispersion

15. Calculate the common dispersion estimates the overall BCV of the dataset, averaged over all genes. [Hint:estimateCommonDisp] (5 points)

16. Calcualte the estimate gene-wise dispersion estimates, allowing a possible trend with averge count siz. [Hint:estimateGLMTrendedDisp, estimateTagwiseDisp] (5 points)

Testing for differential expression

17. Identify top differentially expressed genes between luminal vs basal (10 points)

18. Identify top differentially expressed genes between pregnant and virgin status. (15 points)