Assignment

Use the edgeR package to determine differentially expressed genes using RNA-Seq derived counts.

1.1 Install edgeR

Use the Bioconductor package edgeR to determine differentially expressed genes. Install the package in R:

```
source(“http://bioconductor.org/biocLite.R”)
biocLite( “edgeR”)
```

1.2 Load edgeR and the data file

Load the package within R:

```
library(edgeR)
```

Load the data file (file name stored in fNam) into a data frame and rename the row names of the data frame.

```
cnts <- read.delim(fNam)

row.names(cnts) <- cnts[,"ID"]
```

1.3 Create a DGEList object

The DGEList object is used to store all data. First, the gene expression of strain WCFS1 is determined. The columns 2 till 5 contain the counts of WCFS1 measured for two different growth conditions.

```
exp <- c("WCFS1.glc","WCFS1.glc","WCFS1.rib","WCFS1.rib")

group <- factor(exp)
y <- DGEList(counts=cnts[,2:5],group=group)
```

1.4 Filter the data

Next, genes are selected from the DGEList with at least 50 counts per million (cpm) in two samples. The function cpm of the edgeR package can be used to determine the cpm values.

```
keep.genes <- rowSums(cpm(y)>50) >= 2
y <- y[keep.genes,]
```

After the selection step, the library size needs to be recalculated.

```
y$samples$lib.size <- colSums(y$counts)
```

1.5 Data normalization

The data is normalized between samples by determining scaling factors that minimize the fold changes between samples. The default method is TMM (“trimmed mean of M-values”).

```
y <- calcNormFactors(y, method="TMM" )
```

1.6 Create design matrix

The samples are grouped by conditions into sets using a design matrix.

```
design <- model.matrix(~0+group, data=y$samples)
colnames(design) <- levels(y$samples$group)
print(design)
```

1.7 Estimate dispersion

The counts are used to estimate dispersion values that are used for correcting the distributions. For this, edgeR uses different methods.

```
y <- estimateGLMCommonDisp(y,design)
y <- estimateGLMTrendedDisp(y,design, method="power")
y <- estimateGLMTagwiseDisp(y,design)
```

1.8 Plot normalized data

To check the normalization and dispersion applied to the data, we plot the samples (plotMDS) and the dispersion of the data (plotBCV).

```
pdf("Results.pdf"))
plotMDS(y)
plotBCV(y)
dev.off()
```

1.9 Determine differentially expressed genes

Finally, the normalized counts are used to determine the log fold changes and corrected p-values. Note that the log fold change is defined as "`WCFS1.glc-WCFS1.rib`" which is ^{2}**log**("counts of samples of glucose" / "counts of samples of ribose").

```
fit <- glmFit(y,design)

mc <- makeContrasts(exp.r=WCFS1.glc-WCFS1.rib, levels=design)
fit <- glmLRT(fit, contrast=mc)

res <- topTags(fit)
print(res)
```

The function `topTags` returns the genes with the largest fold changes.