Ocular disease mechanisms elucidated by genetics of human fetal retinal pigment epithelium gene expression

Lab Journal Theme07 - Gene Expression Analysis

Lisa Hu
414264
Bio-Informatica
Hanzehogeschool Groningen, ILST
Marcel Kempenaar
8 March 2022

Contents

1 Loading the data

```
knitr::opts_chunk$set(cache = TRUE)
knitr::opts_chunk$set(echo = TRUE)

# Load packages
packages <- c("pander", "dplyr", "affy", "knitr", "ggplot2", "DESeq2", "pheatmap",
              "PoiClaClu", "scales", "apeglm", "EnhancedVolcano")
invisible(lapply(packages, library, character.only = T))
```

For decompressing the data, run the code chunks in the Rmd file that deem fit for your situation:

- If you downloaded the data from the official site: Decompress the data and run the Rscript `data_loading.R`.
- If you want to use the dataset delivered with the project: Run the `decompress-dataset` code chunk.

```
#' Decompress the complete dataset
#' Use this chunk if you did not download the data from the site and want to use
#'  the delivered gzipped dataset

## Set the count.file variable to the full path of the gene file
count.file <- ""
system(paste("gzip -d", count.file))
```

After decompressing the data, the data can be read:

```
## Read the dataset
dataset <- read.table("./gene_count.txt", sep = "\t", header = TRUE)
## Set rownames of the dataset to first column
row.names(dataset) <- dataset$Gene
## Remove the Gene column
dataset <- dataset[-1]

## Indices for dataset
glucose.data <- seq(1, 48, 2)
galactose.data <- seq(2, 49, 2)
groups <- factor(rep(1:2, times=24), labels = c("Glucose", "Galactose"))
col.ordered <- c(colnames(dataset[glucose.data]), colnames(dataset[galactose.data]))

## Colors for the two sample groups (red = galactose, blue = glucose)
group.cols <- hue_pal()(2)
```

2 Exploratory Data Analysis

2.1 Data sample

```
pander(dataset[0:5, 0:4], split.tables = 64)
```

Table 1: Table continues below

	X1_glucose	X1_galactose	X2_glucose	X2_galactose
__alignment_not_unique	0	0	0	0
__ambiguous	73052	71663	90130	114748
__no_feature	6143654	3901459	4560099	10675855
__not_aligned	0	0	0	0
__too_low_aQual	0	0	0	0

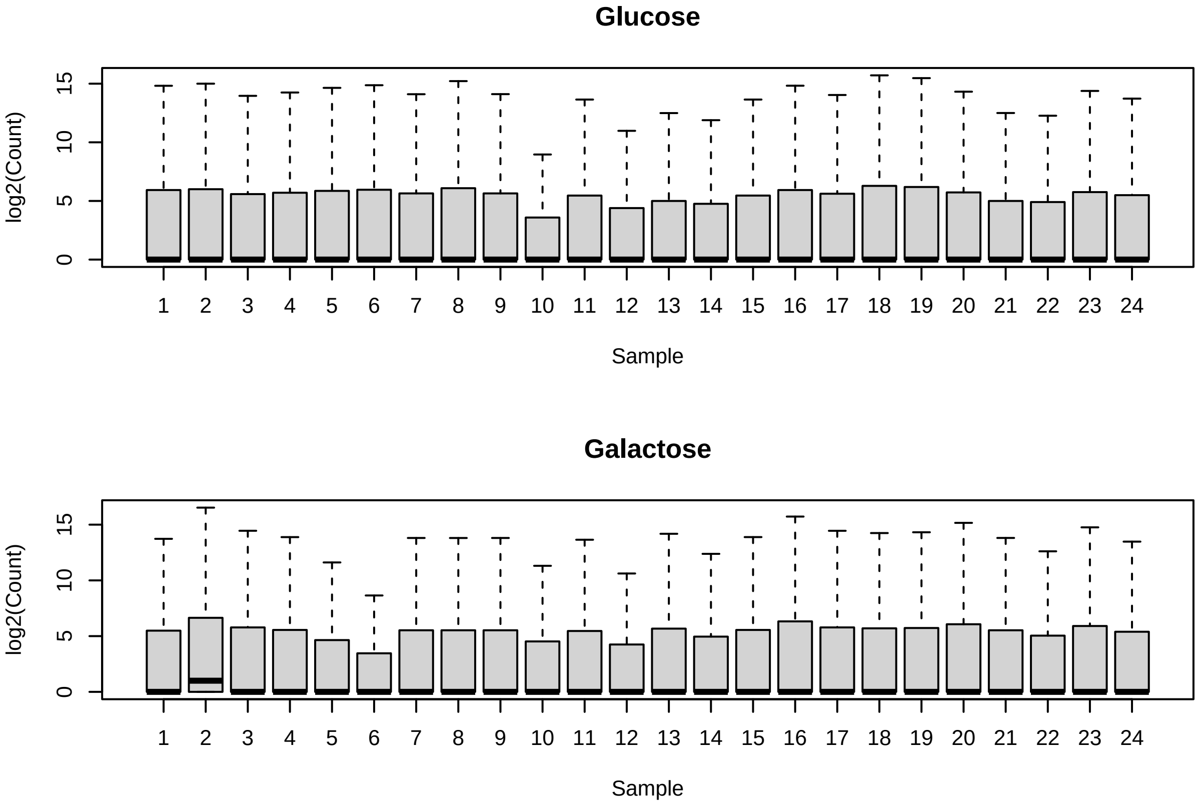
```
pander(summary(dataset[,0:6]), split.tables = 64)
```

Table 3: Table continues below

X1_glucose	X1_galactose	X2_glucose	X2_galactose	X3_glucose	X3_galactose
Min. : 0	Min. : 0	Min. : 0	Min. : 0	Min. : 0	Min. : 0
1st Qu.: 0	1st Qu.: 0	1st Qu.: 0	1st Qu.: 0	1st Qu.: 0	1st Qu.: 0
Median : 0	Median : 0	Median : 0	Median : 1	Median : 0	Median : 0
Mean : 719	Mean : 549	Mean : 750	Mean : 1147	Mean : 622	Mean : 679
3rd Qu.: 60	3rd Qu.: 44	3rd Qu.: 63	3rd Qu.: 99	3rd Qu.: 47	3rd Qu.: 54
Max. :6143654	Max. :3901459	Max. :4560099	Max. :10675855	Max. :5017129	Max. :5650847

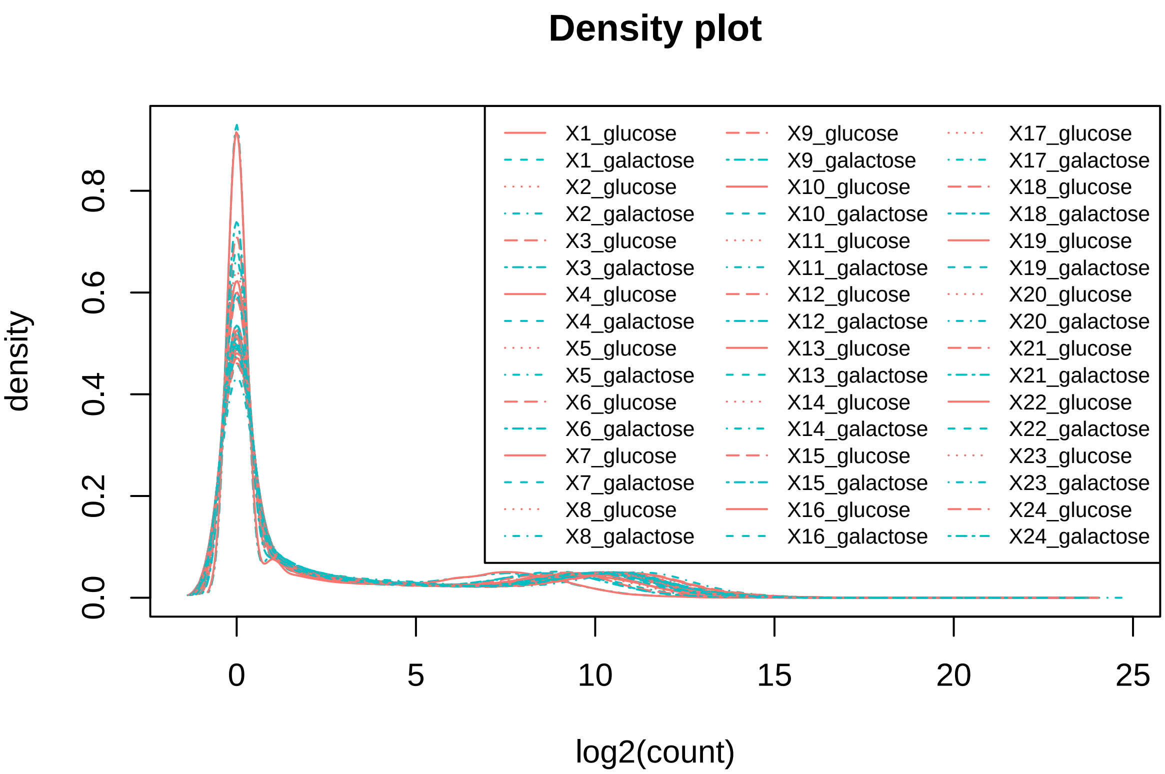
2.2 Boxplots

```
layout(matrix(c(1,1,2,2), nrow = 4, ncol = 1, byrow = T))
## Glucose plot
boxplot(log2(dataset[glucose.data]+1), main = "Glucose", names = seq(1, 24),
        xlab = "Sample", ylab = "log2(Count)", outline = FALSE)
## Galactose plot
boxplot(log2(dataset[galactose.data]+1), main = "Galactose", names = seq(1, 24),
        xlab = "Sample", ylab = "log2(Count)", outline = FALSE)
```



2.3 Density plots

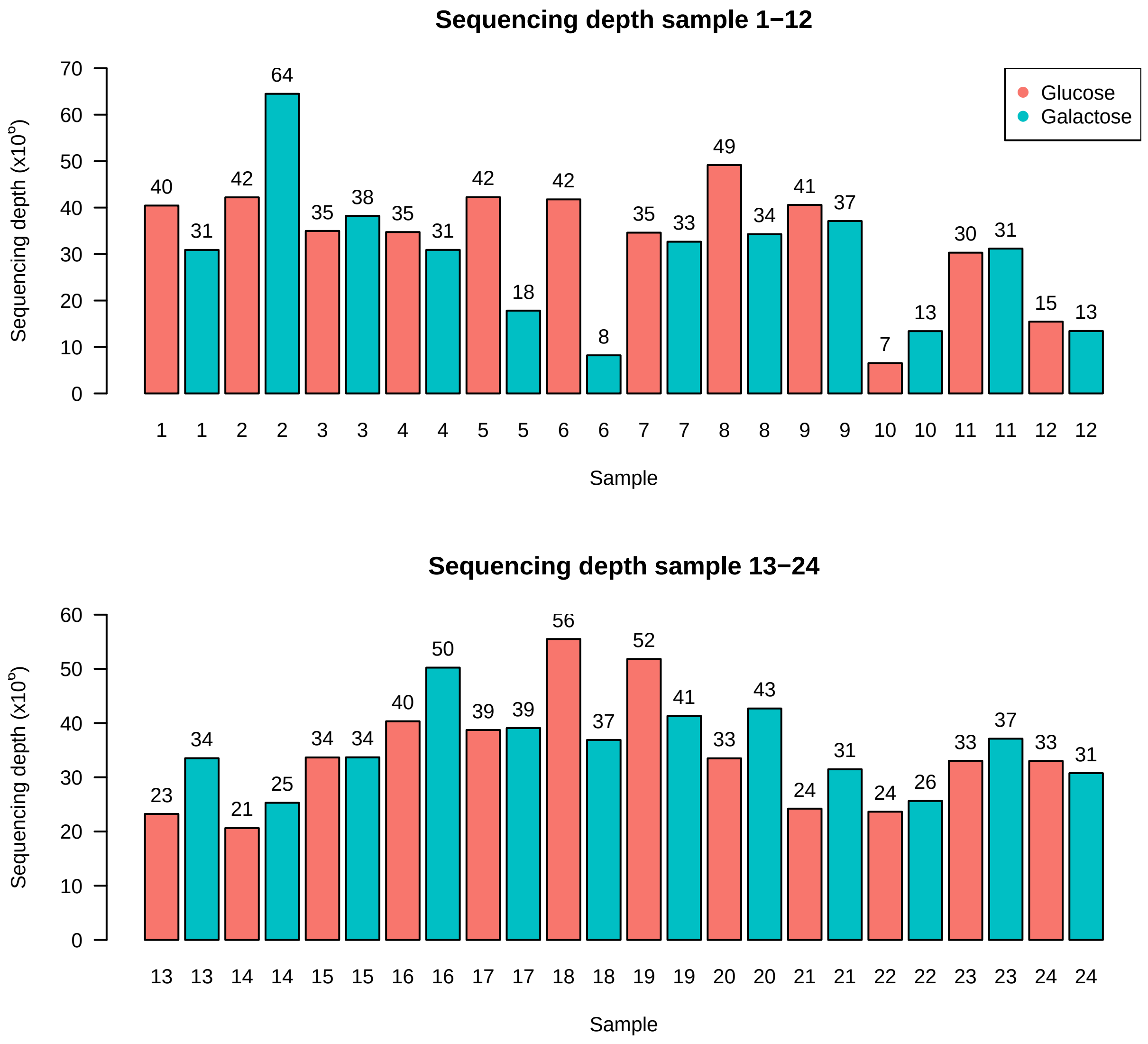
```
plotDensity(log2(dataset+1), main = "Density plot", col = group.cols,
            lty = 1:48, xlab = "log2(count)")
legend("topright", names(dataset), lty = 1:48, col = group.cols,
       cex = 0.7, ncol = 3)
```



2.4 Barplots

```
layout(matrix(c(1,1,1,2,2,2), nrow = 6, ncol = 1, byrow = T))
## Barplot of first half of the data
x1 <- barplot(colSums(dataset[1:24]/ 1e6), main = "Sequencing depth sample 1-12",
              xlab = "Sample", ylab = expression("Sequencing depth (x10"^6*")"),
              ylim = c(0, 70), las = 2, col = group.cols, xaxt = 'n')
text(x = x1, y = colSums(dataset[1:24]/ 1e6),
     label = round(colSums(dataset[1:24]/ 1e6),0), pos = 3)
axis(1, at = x1, labels = rep(1:12, each = 2), tick = FALSE, cex = 0.6)
legend("topright", c("Glucose", "Galactose"), col = group.cols, pch = 19)

## Rest of the data
x2 <- barplot(colSums(dataset[25:48]/ 1e6), main = "Sequencing depth sample 13-24",
              xlab = "Sample", ylab = expression("Sequencing depth (x10"^6*")"),
              ylim = c(0, 60), las = 2, col = group.cols, xaxt = 'n')
text(x = x2, y = colSums(dataset[25:48]/ 1e6),
     label = round(colSums(dataset[25:48]/ 1e6), 0), pos = 3)
axis(1, at = x1, labels = rep(13:24, each = 2), tick = FALSE, cex = 0.6)
```



3 Normalization

```
ddsMat <- DESeqDataSetFromMatrix(countData = round(dataset),
                                 colData = data.frame(samples = names(dataset)),
                                                      design = ~ 1)
rld.dds <- vst(ddsMat)
rld <- assay(rld.dds)
sampledists <- dist(t(rld))
```

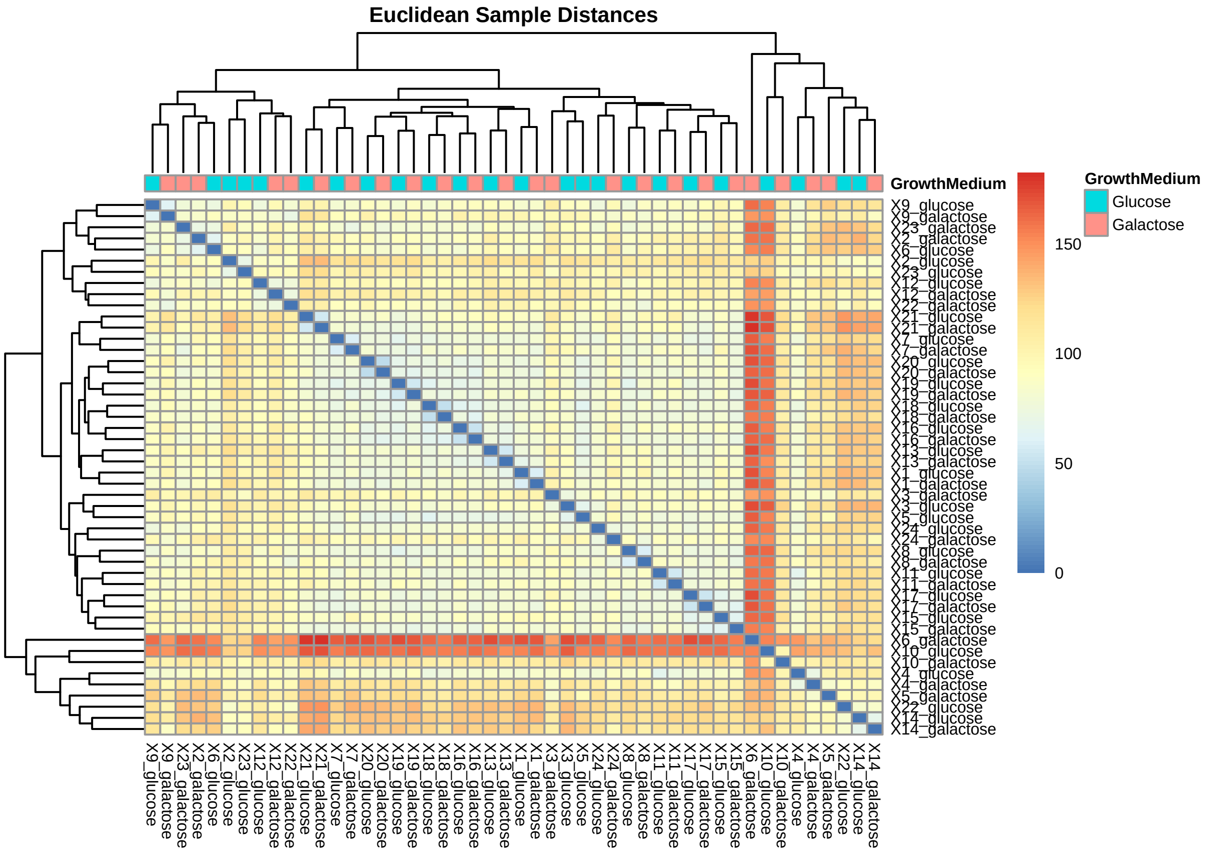
3.1 Heatmaps

```
distMatrix <- as.matrix(sampledists)

annotation <- data.frame(GrowthMedium = groups)

rownames(annotation) <- names(dataset)

pheatmap(distMatrix, show_colnames = T,
         annotation_col = annotation,
         clustering_distance_rows = sampledists,
         clustering_distance_cols = sampledists,
         main = "Euclidean Sample Distances", fontsize= 6)
```



A plot without the clustering and ordered groups:

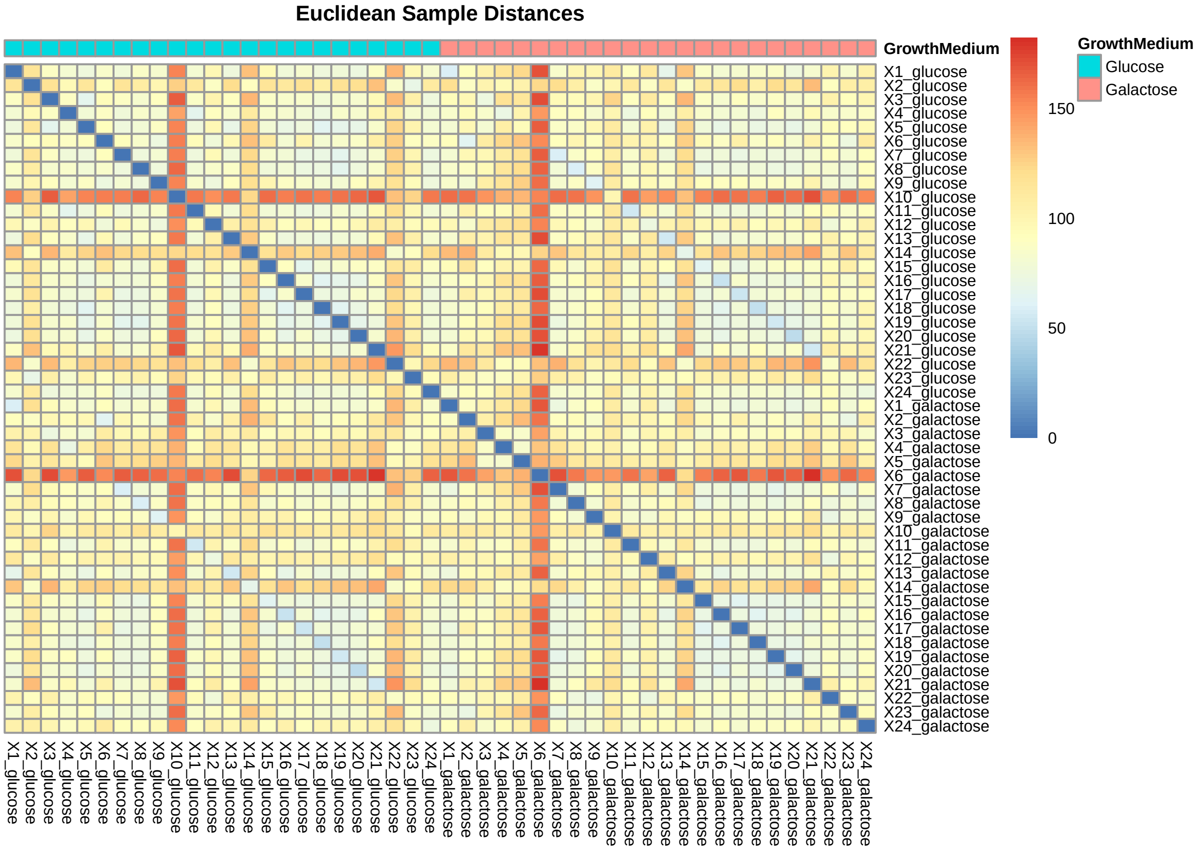
```
rld.ord <- rld[,col.ordered]
sampledists.ord <- dist(t(rld.ord))

distMatrix.ord <- as.matrix(sampledists.ord)

annotation.ord <- data.frame(GrowthMedium = factor(rep(1:2, each = 24),
                                                   labels = c("Glucose", "Galactose")))

rownames(annotation.ord) <- col.ordered

pheatmap(distMatrix.ord, show_colnames = TRUE,
         annotation_col = annotation.ord, cluster_rows = FALSE, cluster_cols = FALSE,
         main = "Euclidean Sample Distances", fontsize= 6)
```

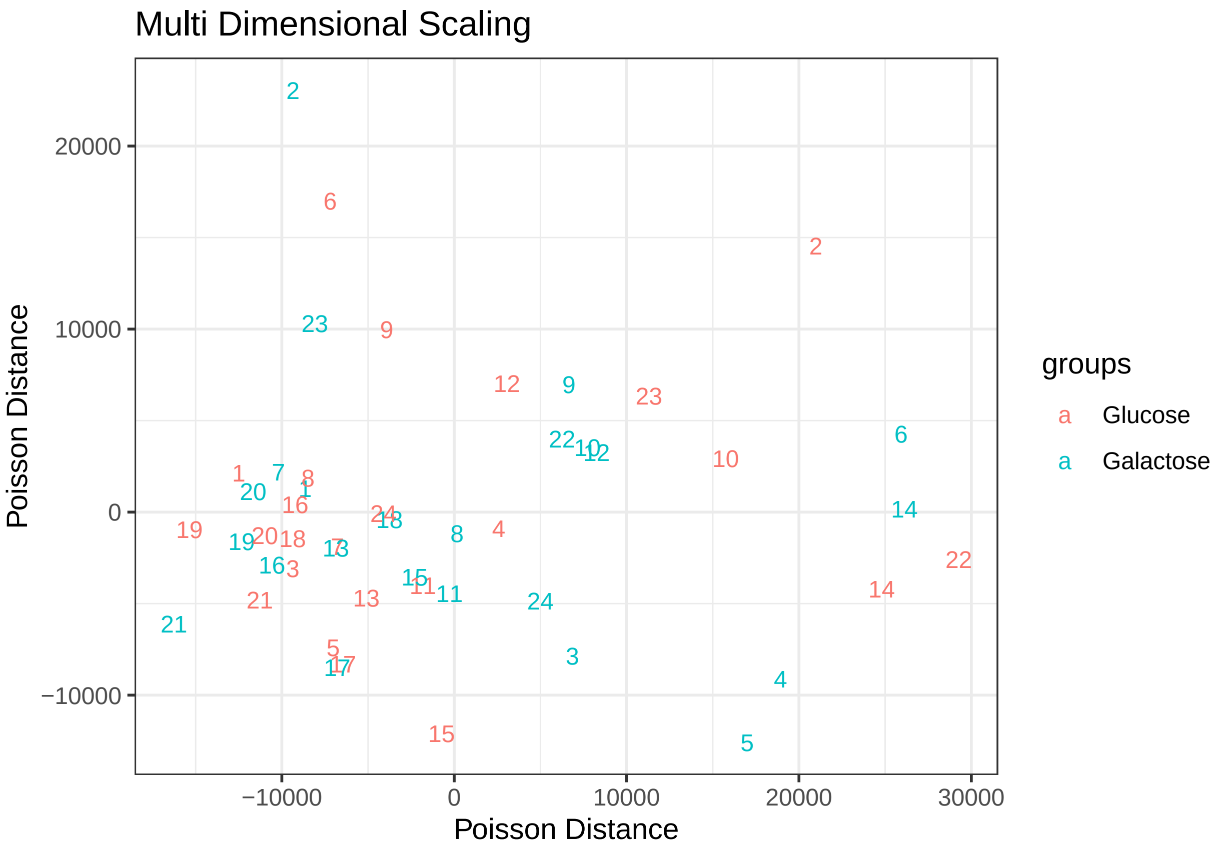


3.2 Multi Dimensional Scaling

Since the colours confirm the sample group, the sample names were simplified to only the number indicating the different samples.

```
dds <- assay(ddsMat)
poisd <- PoissonDistance(t(dds), type="deseq")
## Extract matrix with distances
poisDistMatrix <- as.matrix(poisd$dd)
## Calculate MDS for X- and Y- coordinates
mdsPoisData <- data.frame(cmdscale(poisDistMatrix))
## Readable names
names(mdsPoisData) <- c("x_coord", "y_coord")
## Annotation label
coldata <- rep(1:24, each=2)

ggplot(mdsPoisData, aes(x_coord, y_coord, color = groups, label = coldata)) +
        geom_text(size = 3) +
        ggtitle("Multi Dimensional Scaling") +
        labs(x = "Poisson Distance", y = "Poisson Distance") +
        theme_bw()
```



4 Discovering Differentially Expressed Genes (DEGs)

4.1 Pre-processing

Genes with average read count below 10 and with zero counts in more than 20% of samples (10 samples) were considered not expressed and filtered.

```
fpm <- log2( (dataset / (colSums(dataset) / 1e6)) + 1 )
zero.counts <- apply(dataset, 1, function(x) length(which(x==0)))
filtered.subset <- subset(dataset, rowSums(dataset)/48 > 10 | zero.counts < 10)
```

Leaving the dataset look a bit like this:

```
pander(filtered.subset[0:5, 0:4], split.tables = 64)
```

Table 5: Table continues below

	X1__glucose	X1__galactose	X2__glucose	X2__galactose
___ambiguous	73052	71663	90130	114748
___no_feature	6143654	3901459	4560099	10675855
ENSG00000000419.8	668	613	1015	1141
ENSG00000000457.9	567	494	654	960
ENSG00000000460.12	189	118	146	294

```
pander(summary(filtered.subset[,0:6]), split.tables = 64)
```

Table 7: Table continues below

X1__glucose	X1__galactose	X2__glucose	X2__galactose	X3__glucose	X3__galactose
Min. : 0	Min. : 0	Min. : 0	Min. : 0	Min. : 0	Min. : 0
1st Qu.: 21	1st Qu.: 16	1st Qu.: 22	1st Qu.: 36	1st Qu.: 17	1st Qu.: 20
Median : 317	Median : 242	Median : 349	Median : 526	Median : 259	Median : 284
Mean : 1874	Mean : 1432	Mean : 1956	Mean : 2989	Mean : 1621	Mean : 1771
3rd Qu.: 1500	3rd Qu.: 1180	3rd Qu.: 1659	3rd Qu.: 2524	3rd Qu.: 1281	3rd Qu.: 1357
Max. :6143654	Max. :3901459	Max. :4560099	Max. :10675855	Max. :5017129	Max. :5650847

4.2 DESeq2 analysis

```
## Create design frame
design <- data.frame(groups)
## Create new DDS object with correct design
ddsMat <- DESeqDataSetFromMatrix(countData = filtered.subset,
                                 colData = design, design = ~0 + groups)
ddsMat <- DESeq(ddsMat)
## Results of the new DDS object
dds.res <- results(ddsMat, alpha=0.05)
summary(dds.res)
```

```
##
## out of 21571 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 1034, 4.8%
## LFC < 0 (down)     : 709, 3.3%
## outliers [1]       : 0, 0%
## low counts [2]     : 837, 3.9%
## (mean count < 3)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

```
## Shrinkage
resultsNames(ddsMat)
```

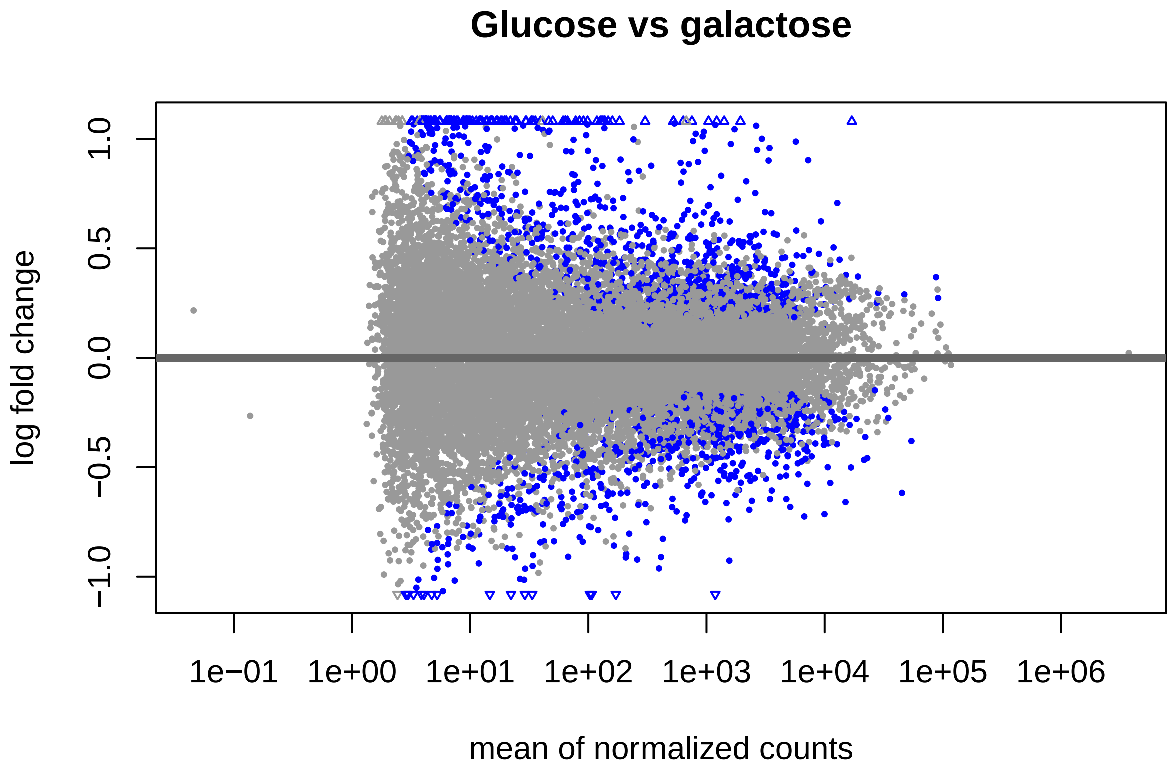
```
## [1] "groupsGlucose"   "groupsGalactose"
```

```
lfc.gal <- lfcShrink(ddsMat, coef = "groupsGalactose", type = "apeglm")
summary(lfc.gal)
```

```
##
## out of 21571 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 21558, 100%
## LFC < 0 (down)     : 0, 0%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 0)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

4.3 MA plot

```r
DESeq2::plotMA(dds.res, main = "Glucose vs galactose")
```


Glucose vs galactose
log fold change
1e−01
1e+00
1e+01
1e+02
1e+03
1e+04
1e+05
1e+06
mean of normalized counts