Zadanie 11 - ocena poprawności analiz (kontrola jakości, mapowanie, DE)

Ksenia Kvitko

25 06 2020

1. Ładowaniebiblioteki

```
library(DESeq2)
```

2. Wczytanie odczytów zmapowanych danych

```
dataERCC92 <- read.delim("~/cwiczenia/zad11/counts_ERCC92.txt", comment.char = "#")
data22 <- read.delim("~/cwiczenia/zad11/counts_22.txt", comment.char = "#")
```

Oraz danych dostarczonych

```
dataProvided <- read.delim("~/cwiczenia/zad11/cms_095046.txt", comment.char = "#")
```

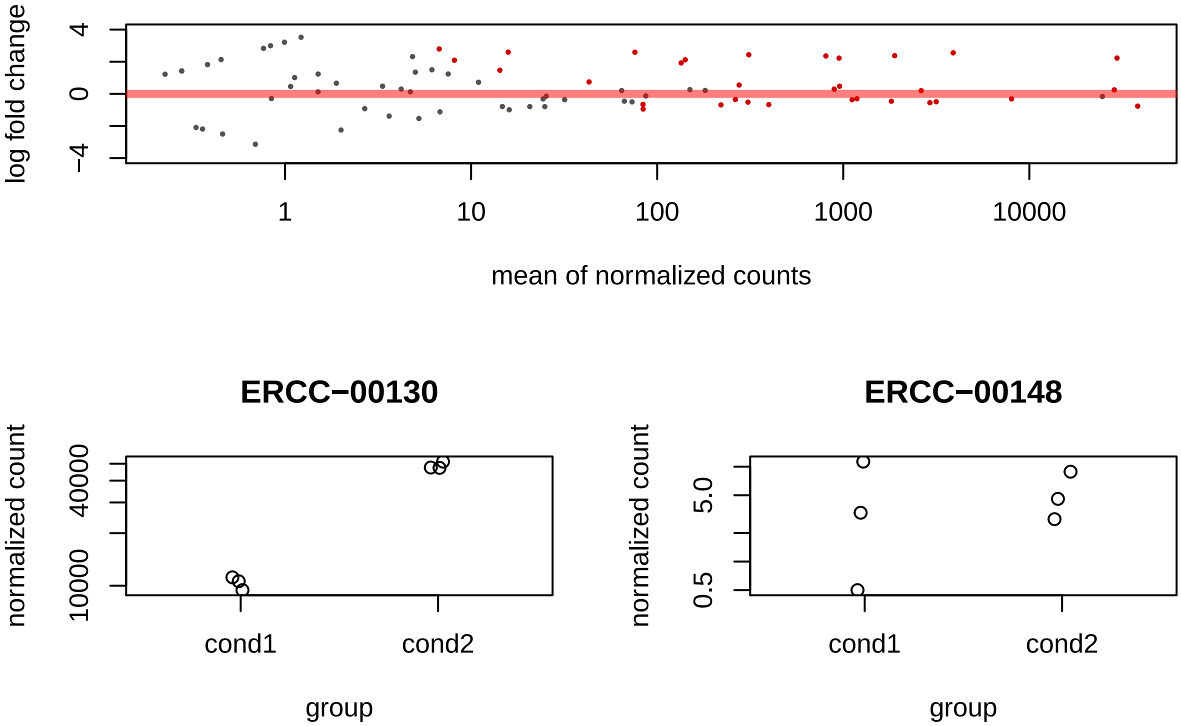
3. Rozwiązanie zadania

3.1 Normalizacja danych

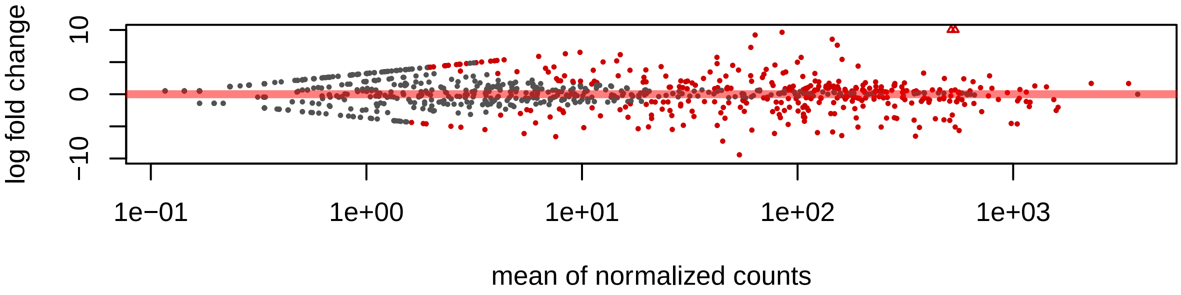
```
normalizeDDS <- function(data){
  countData <- data[,7:12]
  rownames(countData) <- data$Geneid
  samples <- names(countData)
  cond_1 <- rep("cond1", 3)
  cond_2 <- rep("cond2", 3)
  condition <- factor(c(cond_1, cond_2))
  colData <- data.frame(samples = samples, condition = condition)
  dds <- DESeqDataSetFromMatrix(countData = countData, colData = colData, design = ~condition)

  return(dds)
}

normalizedERCC92 <- normalizeDDS(dataERCC92)
normalized22 <- normalizeDDS(data22)
```

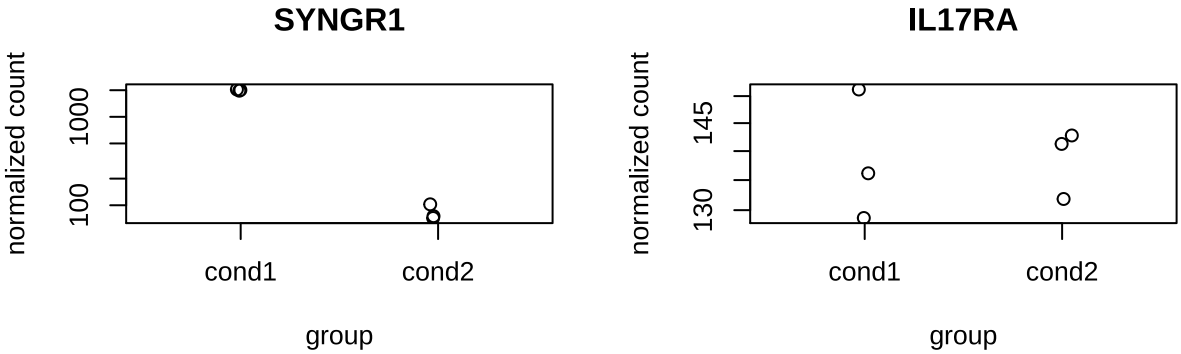
3.2 Analiza DE

```
analyseDE <- function(Data, ymin, ymax){
  dds <- DESeq(Data)
  res <- results(dds)

  layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
  plotMA(res, ylim=c(ymin,ymax))
  plotCounts(dds, gene=which.min(res$padj), intgroup="condition")
  plotCounts(dds, gene=which.max(res$padj), intgroup="condition")

  resOrdered <- res[order(res$padj),]
  resSig <- subset(resOrdered, padj<0.05)
  return(resSig)
}

dddERCC92 <- analyseDE(normalizedERCC92, -4, 4)
```



```
ddd22 <- analyseDE(normalized22, -10, 10)
```





3.3 Ocena korelacji (poprawności analizy)

```
dddERCC92 <- cbind(rownames(dddERCC92), data.frame(dddERCC92, row.names=NULL))
colnames(dddERCC92) <- c("ERCC.ID", "baseMean", "log2FoldChange", "lfcSE", "stat", "pvalue", "padj")

Test.corr <- data.frame(dddERCC92$ERCC.ID, dddERCC92$log2FoldChange)
colnames(Test.corr) <- c("ERCC.ID","log2FoldChange")
rownames(Test.corr) <- Test.corr$ERCC.ID
compare <- data.frame(dataProvided[,2],dataProvided[,7])
colnames(compare) <- c("ERCC.ID","log2")
rownames(compare) <- compare$ERCC.ID

Test.corr <- merge(Test.corr, compare, by=0, all=TRUE)
Test.corr <- na.omit(Test.corr)
rownames(Test.corr) <- Test.corr$Row.names
```

3.3.1 Test korelacji Pearsona

```
corRes <- cor.test(Test.corr$log2, Test.corr$log2FoldChange,method = "pearson", )
corRes
```

```
##
##  Pearson's product-moment correlation
##
## data:  Test.corr$log2 and Test.corr$log2FoldChange
## t = 26.381, df = 26, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9606272 0.9916621
## sample estimates:
```

```
##       cor
## 0.9818287
```

Niskie prawdopodobieństwo (p-value << 0.05) oraz wysoka wartość współczynnika koralacji (cor > 0.98) sugerują poprawne przeprowadzenie analiz.