thema07

keimpe dijkstra

8-3-2022

Exploratory data analysis

imports

```
library(dplyr)
```

```
##
## Attaching package: ’dplyr’

## The following objects are masked from ’package:stats’:
##
##     filter, lag

## The following objects are masked from ’package:base’:
##
##     intersect, setdiff, setequal, union
```

```
library(affy)
```

```
## Loading required package: BiocGenerics

## Loading required package: parallel

##
## Attaching package: ’BiocGenerics’

## The following objects are masked from ’package:parallel’:
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from ’package:dplyr’:
##
##     combine, intersect, setdiff, union

## The following objects are masked from ’package:stats’:
##
##     IQR, mad, sd, var, xtabs
## The following objects are masked from ’package:base’:
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min
```

```
## Loading required package: Biobase
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     ’browseVignettes()’. To cite Bioconductor, see
##     ’citation("Biobase")’, and for packages ’citation("pkgname")’.
```

```r
library(scales)
```

First, we load all our data into r

```r
#load in the data using the read.table method
j147 <- read.table("./data/j147.csv", header = T, sep = ",")
cad31 <- read.table("./data/cad-31.txt", header = T, sep = "\t", fill = T)
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec = dec, :
## EOF within quoted string
```

```r
#create a new column for te gene annotation
cad31$Gene <- strsplit(cad31$Annotation.Divergence, "|", 1)

#filter the gene annotation out of the string
counter <- 1
while( counter < length(cad31$Gene)) {

  cad31$Gene[counter] <- cad31$Gene[[counter]][1]
  counter <- counter + 1
}

#drop some unnecesary columns
cad31 <- cad31[c(22, 9:21)]

#merge all the data into one dataframe
data <- merge(j147, cad31)

#delete old dataframes
rm(cad31)
rm(j147)

#replace na's with zero's
data[is.na(data)] <- 0
```

```
#rename the columns
names(data) <- c(
                paste0('Gene'),
                paste0('AD.old.j147.', 1:3),
                paste0('AD.old.', 1:3),
                paste0('AD.young.', 1:4),
                paste0('AD.cad31.', 1:3),
                paste0('AD.', 1:3),
                paste0('WT.CAD31.', 1:3),
                paste0('WT.', 1:4)
                )

#TODO: Check row names just to be sure

#set some indices to help future work
AD.old.j147 <- 2:4
AD.old <- 5:7
AD.young <- 8:11
AD.cad31 <- 12:14
AD <- 15:17
WT.CAD31 <- 18:20
WT <- 21:24
```

visualizations

basic statistics

```
summary(rowSums(data[AD.old.j147])/3)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     0.33    52.67   569.01   497.17 62003.67
```

```
summary(rowSums(data[AD.old])/3)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     0.33    49.33   531.20   462.67 62088.33
```

```
summary(rowSums(data[AD.young])/4)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0    38.0   437.0   377.9 46680.8
```

```
summary(rowSums(data[AD.cad31])/3)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     0.33    59.33   575.60   470.33 63783.33
```

```
summary(rowSums(data[AD])/3)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##     0.00     0.33    52.17   488.74   406.33 55172.00 
```

```
summary(rowSums(data[WT.CAD31])/3)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##     0.00     0.33    41.33   405.96   339.42 39443.00 
```

```
summary(rowSums(data[WT])/4)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##     0.00     0.50    59.75   624.28   506.06 71066.25 
```

The min and first quantile all show similar or very similar results. The wildtype mouse shows the most expression with an almost double maximun than its drugged counterpart.

The young mouse and the mouse on CAD-31 have the least sequences read whereas the mouse with AD and on drugs seem to be upregulated.

This might be explained because of regulation but also the testing can have have a influence on the amount of sequences that are read, therefore we need to normalize the data.

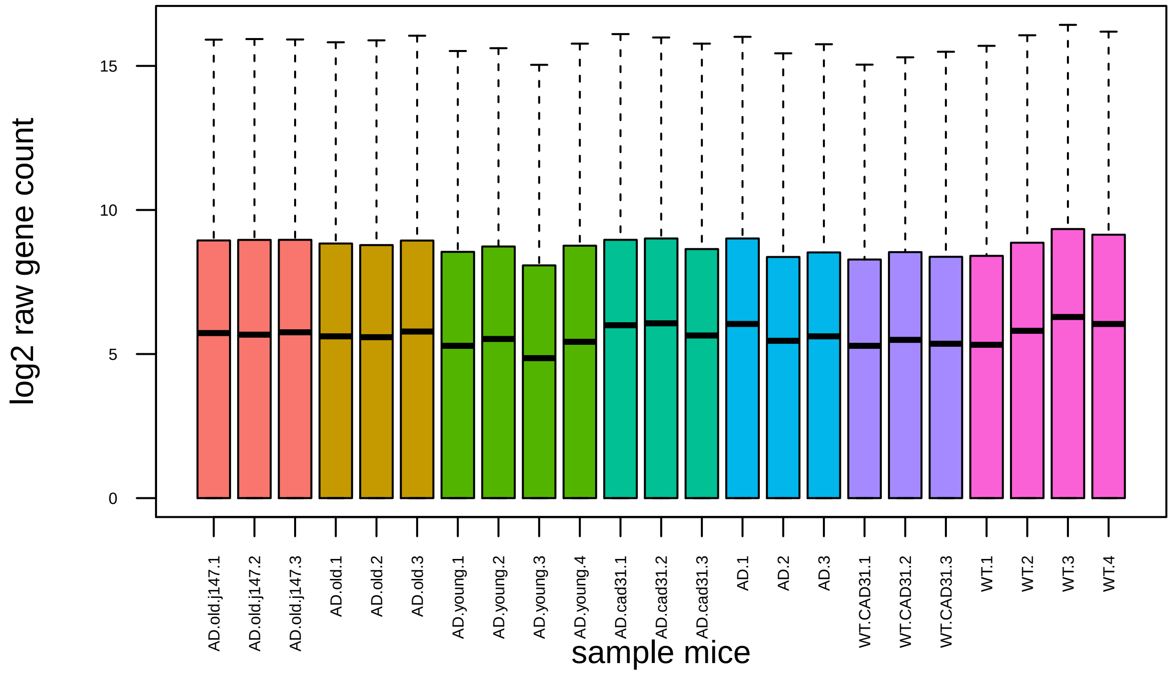
Add 1 to the whole dataframe so it can logscale

```
data <- data[2:24] + 1
```

Boxplot

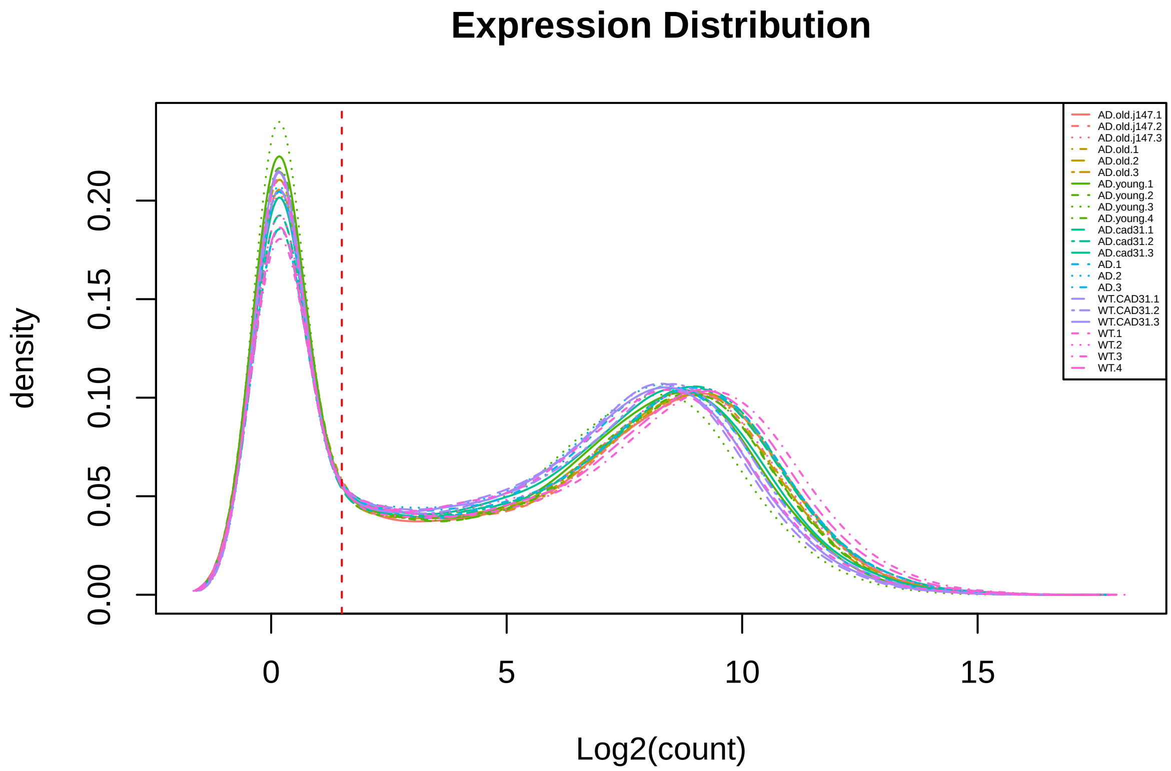
```
myColors <- hue_pal()(7)

boxplot(log2(data),las=2, xlab = "sample mice",
        ylab = "log2 raw gene count", cex.axis=0.5,
        col=rep(myColors,c(3,3,4,3,3,3,4)))
```



Density plot

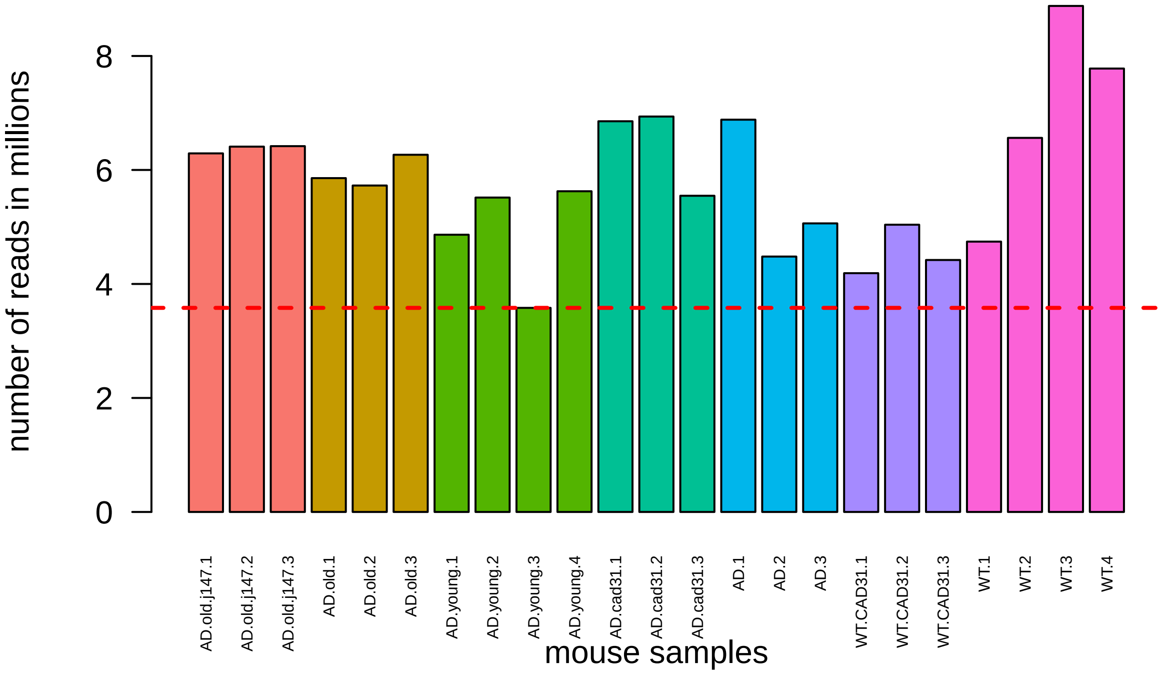
```
## Plot the log2-transformed data with a 0.1 pseudocount
plotDensity(log2(data + 0.1),
            col=rep(myColors,c(3,3,4,3,3,3,4)),
            lty=c(1:ncol(data)), xlab='Log2(count)',
            main='Expression Distribution')
legend('topright', names(data), lty=c(1:ncol(data)),
       col=rep(myColors,c(3,3,4,3,3,3,4)), cex = 0.3)
abline(v=1.5, lwd=1, col='red', lty=2)
```



Barplot sequence depth

```
barplot(colSums(data)/1000000,    col=rep(myColors,c(3,3,4,3,3,3,4)),
        xlab = "mouse samples", ylab = "number of reads in millions", cex.names= 0.5, las=2)

abline(h = min(colSums((data)/1000000)), col = "red",
      lty=2, lwd=2)
```


number of reads in millions
0
2
4
6
8
AD.old.j147.1
AD.old.j147.2
AD.old.j147.3
AD.old.1
AD.old.2
AD.old.3
AD.young.1
AD.young.2
AD.young.3
AD.young.4
AD.cad31.1
AD.cad31.2
AD.cad31.3
AD.1
AD.2
AD.3
WT.CAD31.1
WT.CAD31.2
WT.CAD31.3
WT.1
WT.2
WT.3
WT.4
mouse samples