STAT115 Homework 1

(your name)

2017-01-31

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(knitr)
library(affy)
```

```
celFiles <- list.celfiles(path = "data", full.names=TRUE)
data.affy<-ReadAffy(filenames = celFiles)
data.affy
```

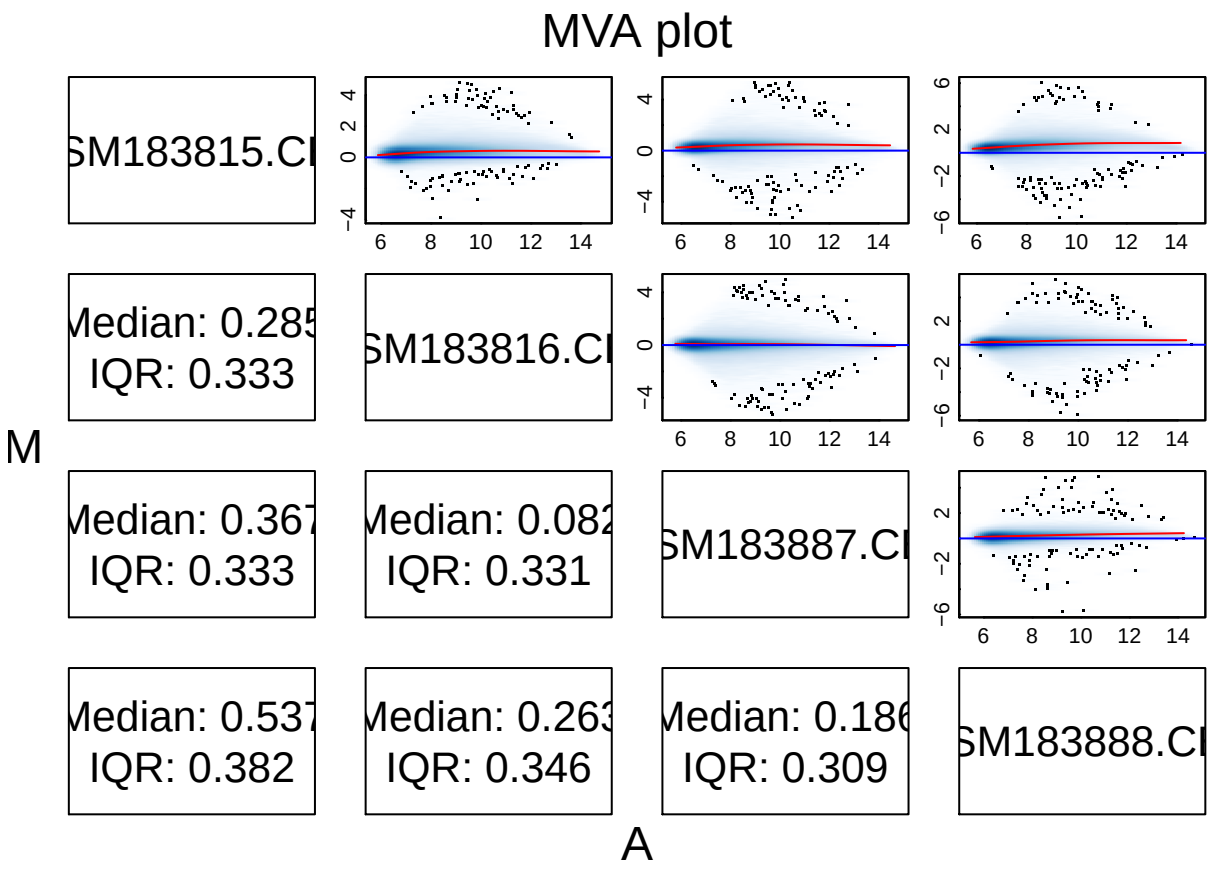
```
## Warning: replacing previous import 'AnnotationDbi::tail' by 'utils::tail'
## when loading 'drosophila2cdf'
```

```
## Warning: replacing previous import 'AnnotationDbi::head' by 'utils::head'
## when loading 'drosophila2cdf'
```

```
##
```

```
## AffyBatch object
## size of arrays=732x732 features (18 kb)
## cdf=Drosophila_2 (18952 affyids)
## number of samples=4
## number of genes=18952
## annotation=drosophila2
## notes=
```

```
MAplot(data.affy, pairs = TRUE, which=c(1,2,3,4), plot.method = "smoothScatter")
```




```
# Raw data visualization
#MA plot for only the first 4 samples - comparing all the samples is too much for one graph
image(data.affy)
hist(data.affy)
boxplot(data.affy, col=c(1:4))

#normalize data
data.rma=rma(data.affy)
expr.rma=exprs(data.rma) #saves expression index as another matrix

#compare normalized data with raw data
library(affyPLM)
MAplot(data.rma, pairs = TRUE, which=c(1,2,3,4), plot.method = "smoothScatter")
boxplot(data.frame(expr.rma), col=seq(2,7,by=1))

#Download tar ball of your cdf file from http://brainarray.mbni.med.umich.edu/Brainarray/Database/Custo
#In practice, you would do this before doing RMA, but in this lab we want to look at the difference bet
library(drosophila2dmentrezgcdf)
data.affy@cdfName="drosophila2dmentrezgcdf"
data.rma.refseq=rma(data.affy)
expr.rma.refseq=exprs(data.rma.refseq)

library(drosophila2dmrefseqcdf)
data.affy@cdfName="drosophila2dmrefseqcdf"
data.rma.refseq=rma(data.affy)
expr.rma.refseq=exprs(data.rma.refseq)
```

```r
#compare row names
rownames(expr.rma)[1:100]
rownames(expr.rma.refseq)[1:100]

#find significant up- and down-regulated genes
control=c(1,2,3,4,5,6)
mutants=c(7,8,9,10,11,12)
genes.refseq=matrix(rownames(expr.rma.refseq))
foldchange=apply(expr.rma, 1, function(x) mean( x[mutants] ) - mean( x[control] ) )
T.p.value=apply(expr.rma, 1, function(x) t.test( x[mutants], x[control], var.equal=T )$p.value )
fdr=p.adjust(T.p.value, method="fdr")
genes.up=genes.refseq[ which( fdr < 0.05 & foldchange > 0 ) ]
genes.down=genes.refseq [ which( fdr < 0.05 & foldchange <0 ) ]

#use limma package to call diff exprs genes
library(limma)
design <- c(0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1) #assign control and trt groups
design.mat <- model.matrix(~design) #to add the intercept term
fit <- lmFit(expr.rma.refseq, design.mat)
fit <- eBayes(fit)
topTable(fit, coef = "design")
```