

Quality control analyses of gene expression data sets from GEO

Warren Anderson
Joon Yuhl Soh
Mete Civelek

March 20, 2020

This guide provides code and documentation of analyses from Anderson et al., 2020, *Sex differences in human adipose tissue gene expression and genetic regulation involve adipogenesis*

Contents

1	Overview	2
2	Quality control analysis of the deCODE data	2
3	Quality control analysis of the AAGMEx data	5
4	References	8

List of Figures

1	Illustration of pairwise sample correlations from the deCODE data set.	3
2	Gene expression distributions from the deCODE data set.	4
3	XIST expression from the deCODE data set.	4
4	Illustration of pairwise sample correlations from the AAGMEx data set.	6
5	Gene expression distributions from the AAGMEx data set.	6
6	XIST expression from the AAGMEx data set.	7

1 Overview

2 Quality control analysis of the deCODE data

We analyzed human subcutaneous adipose data from Icelandic females and males. The data were acquired from the Gene Expression Omnibus (<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE7965>) (Emilsson *et al.*, 2008). The R code below can also be found in the file *Fig1.decodeQC.R*.

```
# load relevant libraries
library(dplyr)
library(reshape2)
library(beeswarm)

# read in data previously imported from GEO
expr_data = read.table("expr_data_decode.txt",header=T,sep="\t",stringsAsFactors=F)
pheno_data = read.table("phenotypes_decode.txt",header=T,sep="\t",stringsAsFactors=F)
genes_data = read.table("genes_decode.txt",header=T,sep="\t",stringsAsFactors=F,
  quote="")
```

First we check if any samples show exaggerated pairwise correlations such that they could potentially be attributed to the same source.

```
# correlation matrix
cor_expr = cor(expr_data)
colnames(cor_expr) = colnames(expr_data)
rownames(cor_expr) = colnames(expr_data)

# corVector function isolates a vector of correlations
# input: correlation matrix
# output: correlation vector
corVector <- function(cor_expr=NULL){
  upperTriangle <- upper.tri(cor_expr, diag=F) # turn into a upper triangle
  cor.upperTriangle <- cor_expr # take a copy of the original cor-mat
  cor.upperTriangle[!upperTriangle] <- NA # set everything not in upper triangle to NA
  cor_melted00 <- melt(cor.upperTriangle, value.name="correlationCoef")
  cor_melted0 <- cor_melted00[!is.na(cor_melted00$correlationCoef),]
  colnames(cor_melted0) <- c("s1", "s2", "cor")
  cor_melted = cor_melted0
  cor_melted[,1:2] = apply(cor_melted[,1:2],2,as.character)
  cor_melted[,3] = unlist(sapply(cor_melted[,3],as.numeric))
  return(cor_melted)
}

# isolate vector of correlations
cor_melted = corVector(cor_expr)
max(cor_melted$cor)

# look for discontinuities in the correlation distributions
# this might indicate duplicated or mixed samples
pdf("decode_sample_correlations.pdf",width=5,height=4)
par(mfrow=c(1,2))
hist(cor_melted$cor,xlab="Pearson correlation",main="",cex.lab=1.2)
hist(cor_melted$cor,xlim=c(0.8,1),ylim=c(0,50),breaks=seq(-1,1,0.001),
  xlab="Pearson correlation",main="",cex.lab=1.2)
abline(v=0.9,col="red")
dev.off()
```

The plot generated is shown in Figure 1. The left panel shows the full distribution of correlation coefficients and the right panel shows an expanded view of the rightmost tail of the distribution. The

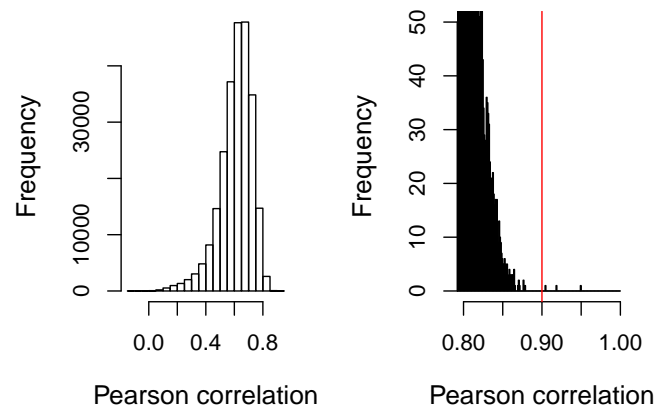


Figure 1: Illustration of pairwise sample correlations from the deCODE data set.

smooth distribution is interrupted and a few isolated correlations are observed beyond 0.9, as indicated by the red vertical line. We removed samples from each pairwise correlation exceeding 0.9 for further analysis.

```
# remove questionable high correlations
ind = which(cor_melted$cor > 0.9)
samp_rem = as.vector( c(cor_melted$s1[ind], cor_melted$s2[ind]) )
ind_rem = sapply(samp_rem,function(x){which(colnames(expr_data)==x)}) %>% unlist
expr_data1 = expr_data[,-ind_rem]

# remove corresponding samples from sex annotation
pheno_data1 = pheno_data[-ind_rem,]

# check
all(pheno_data1$sample == names(expr_data1))
table(pheno_data1$sex) # female: 400 male: 295
```

Next we verify that the expression data are in the log space by plotting the expression distributions for random genes (Figure 2). The data suggest that the expression values are in the log space.

```
# plot expression histograms for three random genes
pdf("decode_gene_dist.pdf",height=3)
par(mfrow=c(1,3))
hist(expr_data1[1,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
hist(expr_data1[100,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
hist(expr_data1[1000,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
dev.off()
```

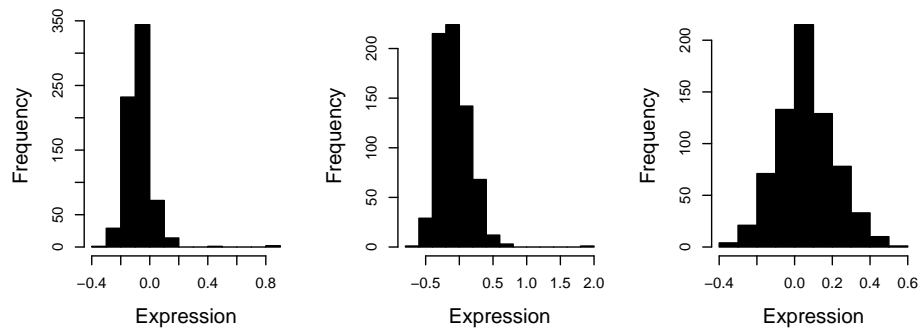


Figure 2: Gene expression distributions from the deCODE data set.

Finally, we verify sex annotation by plotting XIST expression with respect to sex. As shown in Figure 3, the females have substantially greater XIST expression.

```
# plot XIST
pdf("decode_XIST_mf.pdf",width=4,height=4)
indXIST = which(genes_data$gene_list=="XIST")
dat_xist = expr_data1[indXIST,] %>% t
pltdat = cbind(dat_xist,pheno_data1$sex) %>% as.data.frame
names(pltdat) = c("XIST", "sex")
pltdat$XIST = pltdat$XIST %>% data.matrix %>% as.numeric
beeswarm(XIST~sex, data = pltdat, cex=0.25, pch=16, col=c("red","blue"),
  cex.lab=1.3, xlab="")
dev.off()
```

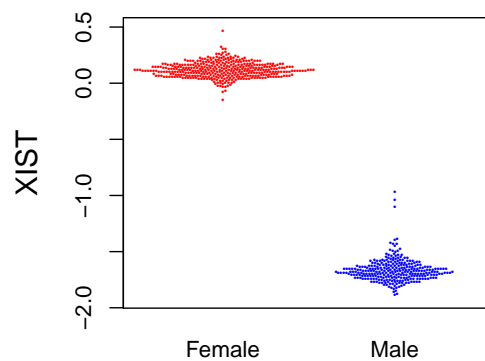


Figure 3: XIST expression from the deCODE data set.

The deCODE data are written to file for subsequent analysis.

```
# write data to file
write.table(expr_data1,"expr_data_decode_qc.txt",col.names=T,row.names=T,
  quote=F,sep="\t")
write.table(pheno_data1,"phenotypes_decode_qc.txt",col.names=T,row.names=F,
  quote=F,sep="\t")
```

3 Quality control analysis of the AAGMEx data

We analyzed human subcutaneous adipose data from African American females and males. The data were acquired from the Gene Expression Omnibus (<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE95674>) (Sharma *et al.*, 2016). The R code below are in *Fig1.aagmexQC.R*.

```
# load relevant libraries
library(dplyr)
library(reshape2)
library(beeswarm)

# read in data previously imported from GEO
expr_data = read.table("expr_data_aagmex.txt",header=T,sep="\t",stringsAsFactors=F)
pheno_data = read.table("phenotypes_aagmex.txt",header=T,sep="\t",stringsAsFactors=F)
```

First we check if any samples show exaggerated pairwise correlations such that they could potentially be attributed to the same source.

```
# correlation matrix
cor_expr = cor(expr_data)
colnames(cor_expr) = colnames(expr_data)
rownames(cor_expr) = colnames(expr_data)

# isolate vector of correlations
cor_melted = corVector(cor_expr)
max(cor_melted$cor)

# look for discontinuities in the correlation distributions
# this might indicate duplicated or mixed samples
pdf("aagmex_sample_correlations.pdf",width=5,height=4)
par(mfrow=c(1,2))
hist(cor_melted$cor,xlab="Pearson correlation",main="",cex.lab=1.2)
hist(cor_melted$cor,xlim=c(0.98,1),ylim=c(0,50),breaks=seq(-1,1,0.001),
      xlab="Pearson correlation",main="",cex.lab=1.2)
abline(v=0.995,col="red")
dev.off()
```

As shown in Figure 4, the distribution of correlation coefficients indicated high sample-to-sample correlations (mean=0.978, compare with Figure 1). The right tail of the distribution decayed smoothly with only one apparent outlier to the right of the red vertical line, for which the associated samples were removed.

```
# remove questionable high correlations
ind = which(cor_melted$cor > 0.995)
samp_rem = as.vector( c(cor_melted$s1[ind], cor_melted$s2[ind]) )
ind_rem = sapply(samp_rem,function(x)which(colnames(expr_data)==x)) %>% unlist
expr_data1 = expr_data[,-ind_rem]

# remove corresponding samples from sex annotation
pheno_data1 = pheno_data[-ind_rem,]

# check
all(pheno_data1$sample == names(expr_data1))
```

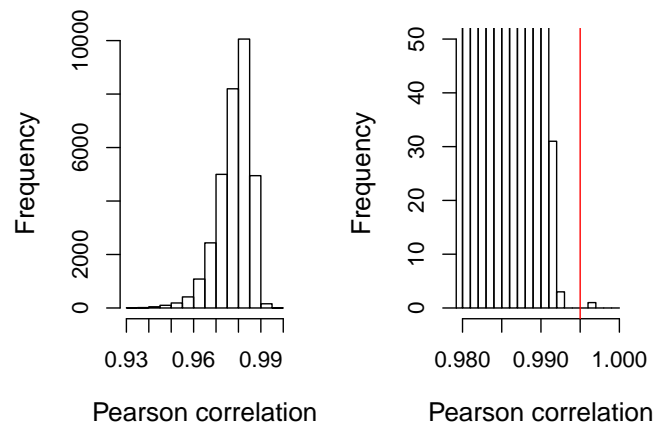


Figure 4: Illustration of pairwise sample correlations from the AAGMEX data set.

Next we verify that the expression data are in the log space by plotting the expression distributions for random genes (Figure 5). The data suggest that the expression values are in the log space.

```
# plot expression histograms for three random genes
pdf("aagmex_gene_dist.pdf",height=3)
par(mfrow=c(1,3))
hist(expr_data1[1,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
hist(expr_data1[100,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
hist(expr_data1[1000,] %>% data.matrix %>% as.numeric,col="black",
      main="",xlab="Expression",cex.lab=1.4)
dev.off()
```

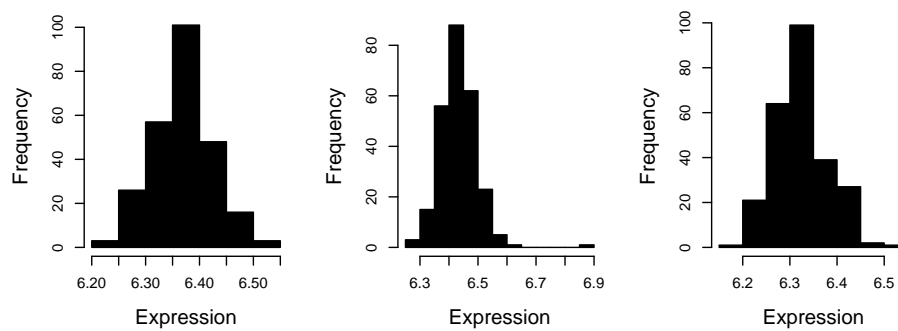


Figure 5: Gene expression distributions from the AAGMEX data set.

Finally, we verify sex annotation by plotting XIST expression with respect to sex. As shown in Figure 6, the females have substantially greater XIST expression. However, two samples had relatively elevated XIST expression, which were above the horizontal line, and we removed these samples for further analysis.

```
# plot XIST
pheno_data1$sex[pheno_data1$sex=="M"] = "Male"
pheno_data1$sex[pheno_data1$sex=="F"] = "Female"
pdf("aagmex_XIST_mf.pdf",width=4,height=4)
indXIST = which(rownames(expr_data1)=="XIST")
dat_xist = expr_data1[indXIST,] %>% t
pltdat = cbind(dat_xist,pheno_data1$sex) %>% as.data.frame
names(pltdat) = c("XIST", "sex")
pltdat$XIST = pltdat$XIST %>% data.matrix %>% as.numeric
beeswarm(XIST~sex, data = pltdat, cex=0.25, pch=16, col=c("red","blue"),
         cex.lab=1.3, xlab="")
abline(h=8)
dev.off()

# remove male samples with high XIST
cut = 8
M_remove = dat_xist[which(dat_xist[pheno_data1$sex=="Male"] > cut),] %>% names
ind_remove = sapply(M_remove,function(x)which(colnames(expr_data1)==x)) %>% unlist
expr_data2 = expr_data1[,-ind_remove]
pheno_data2 = pheno_data1[-ind_remove,]

# check
all(names(expr_data2) == pheno_data2$accession)
table(pheno_data2$sex) # female: 117 male: 135
```

The AAGMEx data are written to file for subsequent analysis.

```
# write data to file
write.table(expr_data2,"expr_data_aagmex_qc.txt",col.names=T,row.names=T,
           quote=F,sep="\t")
write.table(pheno_data2,"phenotypes_aagmex_qc.txt",col.names=T,row.names=F,
           quote=F,sep="\t")
```

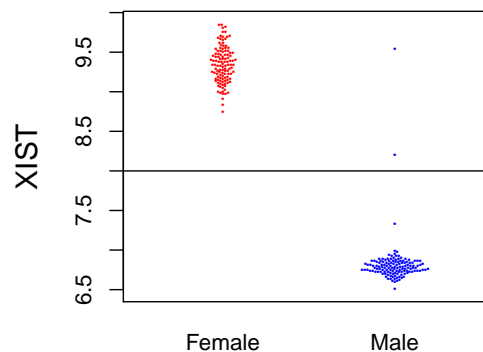


Figure 6: XIST expression from the AAGMEx data set.

4 References

- Emilsson V, Thorleifsson G, Zhang B, Leonardson AS, Zink F, Zhu J, Carlson S, Helgason A, Walters GB, Gunnarsdottir S, Mouy M, Steinthorsdottir V, Eiríksdóttir GH, Björnsdóttir G, Reynisdóttir I, Gudbjartsson D, Helgadóttir A, Jonasdóttir A, Jonasdóttir A, Styrkarsdóttir U, Gretarsdóttir S, Magnusson KP, Stefansson H, Fossdal R, Kristjansson K, Gislason HG, Stefansson T, Leifsson BG, Thorsteinsdóttir U, Lamb JR, Gulcher JR, Reitman ML, Kong A, Schadt EE, Stefansson K (2008). "Genetics of gene expression and its effect on disease." *Nature*, **452**, 423–428.
- Sharma NK, Sajuthi SP, Chou JW, Calles-Escandon J, Demons J, Rogers S, Ma L, Palmer ND, McWilliams DR, Beal J, Comeau ME, Cherry K, Hawkins GA, Menon L, Kouba E, Davis D, Burris M, Byerly SJ, Easter L, Bowden DW, Freedman BI, Langefeld CD, Das SK (2016). "Tissue-Specific and Genetic Regulation of Insulin Sensitivity-Associated Transcripts in African Americans." *The Journal of Clinical Endocrinology and Metabolism*, **101**, 1455–1468.