**Giovanni Coppola Laboratory, UCLA**

**This analysis is performed with software R version 3.4.2**

<https://www.r-project.org/>

**##load packages**

library("gplots")

library("heatmap.plus")

library("MASS")

library(edgeR)

library(DESeq)

library(limma)

library("RColorBrewer")

#To install bioconductor packages

source("https://bioconductor.org/biocLite.R")

biocLite("limma")

#To install R packages

install.packages("gplots")

**##set work directory**

setwd("F:/STUDY/2017-9162\_SwapnaJoshi\_quanSeq/DEA\_edgeR")

**##read in count**

rawCount<-read.csv("metaReadCount\_subset.csv")

dim(rawCount)

rownames(rawCount)<- rawCount$gene\_id

annot<- rawCount[,1:5]

cc<- rawCount[,6:138]

**##read in phenodata**

targets<-read.csv("targets.csv")

labls<-paste(targets$SampleName, targets$Dx, targets$Gender, targets$BH\_Colon\_Exam, sep="\_")

colnames(cc)<-labls

**##create DGEList**

y<- DGEList(counts=cc, genes=annot)

**# filter out low expressed genes**

keep <- rowSums(cpm(y)>1) >= 36

y <- y[keep, keep.lib.sizes=FALSE]

dim(y)

**## plot density with logCPM**

dx<-ifelse(targets$Dx =="HC","orange","cyan")

col<-dx

pdf("1\_density\_0.5cpm.pdf",width=14,height=7)

nsamples <- ncol(y)

par(mfrow=c(1,2))

lcpm0 <- cpm(y0, log=TRUE)

plot(density(lcpm0[,1]), col=col[1], lwd=2, ylim=c(0,2), las=2,

main="", xlab="")

title(main=" Raw data", xlab="Log-cpm")

abline(v=0, lty=3)

for (i in 2:nsamples){

den <- density(lcpm0[,i])

lines(den$x, den$y, col=col[i], lwd=2)

}

legend(x="topright",legend=unique(targets$Dx),col=unique(dx),fill= unique(dx), cex=1)

lcpm <- cpm(y, log=TRUE)

plot(density(lcpm[,1]), col=col[1], lwd=2, ylim=c(0,0.5), las=2,

main="", xlab="")

title(main=" Filtered data", xlab="Log-cpm")

abline(v=0, lty=3)

for (i in 2:nsamples){

den <- density(lcpm[,i])

lines(den$x, den$y, col=col[i], lwd=2)

}

legend(x="topright",legend=unique(targets$Dx),col=unique(dx),fill= unique(dx), cex=1)dev.off()

**##boxplot of expression with logCPM**

x2<-y

x2$samples$norm.factors <- 1

yy <- calcNormFactors(y, method="TMM")

pdf("2\_boxplot.pdf",width=15,height=8)

par(mfrow=c(1,2))

par(cex.axis=0.7)

lcpm <- cpm(x2, log=TRUE)

boxplot(lcpm, las=2, col= dx, main="")

title(main="Filtered and Unnormalized data",ylab="Log-cpm")

legend(x="topright",legend=unique(targets$Dx),col=unique(dx),fill= unique(dx), cex=1)

lcpmn <- cpm(yy, log=TRUE)

par(cex.axis=0.7)

boxplot(lcpmn, las=2, col= dx, main="")

title(main=" Filtered and Normalized data",ylab="Log-cpm")

legend(x="topright",legend=unique(targets$Dx),col=unique(dx),fill= unique(dx), cex=1)

dev.off()

**##design matrix**

TS<- paste(targets$Dx, targets$Gender, targets$BH\_Colon\_Exam, sep="\_")

lane<-as.factor(targets$Lane)

TS <- factor(TS, levels=unique(TS))

design <- model.matrix(~0+TS+lane)

colnames(design)[1:8]<-c( "HC\_F\_N", "IBS\_F\_M", "IBS\_F\_D", "HC\_M\_N", "IBS\_M\_D", "IBS\_M\_M", "IBS\_F\_C", "IBS\_M\_C")

my.contrasts<- makeContrasts( HC.M.N\_vs\_HC.F.N = HC\_M\_N - HC\_F\_N,

IBS.M.C\_vs\_IBS.F.C = IBS\_M\_C - IBS\_F\_C,

IBS.M.D\_vs\_IBS.F.D = IBS\_M\_D - IBS\_F\_D,

IBS.M.M\_vs\_IBS.F.M = IBS\_M\_M - IBS\_F\_M,

IBS.M\_vs\_IBS.F = (IBS\_M\_M+ IBS\_M\_D+ IBS\_M\_C)/3-(IBS\_F\_M+ IBS\_F\_D+ IBS\_F\_C)/3,

IBS.M\_vs\_HC.M=(IBS\_M\_M+ IBS\_M\_D+ IBS\_M\_C)/3 - HC\_M\_N,

IBS.F\_vs\_HC.F=(IBS\_F\_M+ IBS\_F\_D+ IBS\_F\_C)/3 - HC\_F\_N,

IBS\_vs\_HC=( IBS\_M\_M+ IBS\_M\_D+ IBS\_M\_C+ IBS\_F\_M+ IBS\_F\_D+ IBS\_F\_C)/6 - (HC\_M\_N + HC\_F\_N)/2,

IBS.M.C\_vs\_HC.M.N = IBS\_M\_C - HC\_M\_N,

IBS.F.C\_vs\_HC.F.N = IBS\_F\_C - HC\_F\_N,

IBS.M.D\_vs\_HC.M.N = IBS\_M\_D - HC\_M\_N,

IBS.F.D\_vs\_HC.F.N = IBS\_F\_D - HC\_F\_N,

IBS.M.M\_vs\_HC.M.N = IBS\_M\_M - HC\_M\_N,

IBS.F.M\_vs\_HC.F.N = IBS\_F\_M - HC\_F\_N,

levels=design)

**##mean-variance trend**

pdf("2\_voom\_qn.pdf", height=10, width=10)

v <- voom(y,design,plot=TRUE, normalize="quantile")

dev.off()

**##normalize counts**

yy <- calcNormFactors(y, method="TMM")

**##estimate common dispersion**

yy <- estimateGLMCommonDisp(yy,design, verbose=TRUE)

**##estimate trend dispersion**

yy <- estimateGLMTrendedDisp(yy,design)

**##estimate tagwise dispersion**

yy <- estimateGLMTagwiseDisp(yy, design)

**##fit model**

fit <- glmFit(yy,design)

**##contrast**

##contrast 1

lrt1 <- glmLRT(fit, contrast=my.contrasts[,"HC.M.N\_vs\_HC.F.N"])

lrt1$table <- cbind(lrt1$table, FDR=p.adjust(lrt1$table$PValue,method="BH"))

names(lrt1$table) <- paste(names(lrt1$table),"HC.M.N\_vs\_HC.F.N", sep="\_")

lrt1.out<-lrt1$table

##contrast 2

lrt2 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.C\_vs\_IBS.F.C"])

lrt2$table <- cbind(lrt2$table, FDR=p.adjust(lrt2$table$PValue,method="BH"))

names(lrt2$table) <- paste(names(lrt2$table), "IBS.M.C\_vs\_IBS.F.C", sep="\_")

lrt2.out<-lrt2$table

##contrast 3

lrt3 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.D\_vs\_IBS.F.D"])

lrt3$table <- cbind(lrt3$table, FDR=p.adjust(lrt3$table$PValue,method="BH"))

names(lrt3$table) <- paste(names(lrt3$table), "IBS.M.D\_vs\_IBS.F.D",sep="\_")

lrt3.out<-lrt3$table

##contrast 4

lrt4 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.M\_vs\_IBS.F.M"])

lrt4$table <- cbind(lrt4$table, FDR=p.adjust(lrt4$table$PValue,method="BH"))

names(lrt4$table) <- paste(names(lrt4$table), "IBS.M.M\_vs\_IBS.F.M",sep="\_")

lrt4.out<-lrt4$table

##contrast 5

lrt5 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M\_vs\_IBS.F"])

lrt5$table <- cbind(lrt5$table, FDR=p.adjust(lrt5$table$PValue,method="BH"))

names(lrt5$table) <- paste(names(lrt5$table), "IBS.M\_vs\_IBS.F",sep="\_")

lrt5.out<-lrt5$table

##contrast 6

lrt6 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M\_vs\_HC.M"])

lrt6$table <- cbind(lrt6$table, FDR=p.adjust(lrt6$table$PValue,method="BH"))

names(lrt6$table) <- paste(names(lrt6$table), "IBS.M\_vs\_HC.M",sep="\_")

lrt6.out<-lrt6$table

##contrast 7

lrt7 <- glmLRT(fit, contrast=my.contrasts[,"IBS.F\_vs\_HC.F"])

lrt7$table <- cbind(lrt7$table, FDR=p.adjust(lrt7$table$PValue,method="BH"))

names(lrt7$table) <- paste(names(lrt7$table), "IBS.F\_vs\_HC.F",sep="\_")

lrt7.out<-lrt7$table

##contrast 8

lrt8 <- glmLRT(fit, contrast=my.contrasts[,"IBS\_vs\_HC"])

lrt8$table <- cbind(lrt8$table, FDR=p.adjust(lrt8$table$PValue,method="BH"))

names(lrt8$table) <- paste(names(lrt8$table), "IBS\_vs\_HC",sep="\_")

lrt8.out<-lrt8$table

##contrast 9

lrt9 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.C\_vs\_HC.M.N"])

lrt9$table <- cbind(lrt9$table, FDR=p.adjust(lrt9$table$PValue,method="BH"))

names(lrt9$table) <- paste(names(lrt9$table), "IBS.M.C\_vs\_HC.M.N",sep="\_")

lrt9.out<-lrt9$table

##contrast 10

lrt10 <- glmLRT(fit, contrast=my.contrasts[,"IBS.F.C\_vs\_HC.F.N"])

lrt10$table <- cbind(lrt10$table, FDR=p.adjust(lrt10$table$PValue,method="BH"))

names(lrt10$table) <- paste(names(lrt10$table), "IBS.F.C\_vs\_HC.F.N ",sep="\_")

lrt10.out<-lrt10$table

##contrast 11

lrt11 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.D\_vs\_HC.M.N"])

lrt11$table <- cbind(lrt11$table, FDR=p.adjust(lrt11$table$PValue,method="BH"))

names(lrt11$table) <- paste(names(lrt11$table), "IBS.M.D\_vs\_HC.M.N",sep="\_")

lrt11.out<-lrt11$table

##contrast 12

lrt12 <- glmLRT(fit, contrast=my.contrasts[,"IBS.F.D\_vs\_HC.F.N"])

lrt12$table <- cbind(lrt12$table, FDR=p.adjust(lrt12$table$PValue,method="BH"))

names(lrt12$table) <- paste(names(lrt12$table), "IBS.F.D\_vs\_HC.F.N",sep="\_")

lrt12.out<-lrt12$table

##contrast 13

lrt13 <- glmLRT(fit, contrast=my.contrasts[,"IBS.M.M\_vs\_HC.M.N"])

lrt13$table <- cbind(lrt13$table, FDR=p.adjust(lrt13$table$PValue,method="BH"))

names(lrt13$table) <- paste(names(lrt13$table), "IBS.M.M\_vs\_HC.M.N",sep="\_")

lrt13.out<-lrt13$table

##contrast 14

lrt14 <- glmLRT(fit, contrast=my.contrasts[,"IBS.F.M\_vs\_HC.F.N"])

lrt14$table <- cbind(lrt14$table, FDR=p.adjust(lrt14$table$PValue,method="BH"))

names(lrt14$table) <- paste(names(lrt14$table), "IBS.F.M\_vs\_HC.F.N",sep="\_")

lrt14.out<-lrt14$table

#

lrtTot <- as.data.frame(cbind(y$genes,lrt1.out, lrt2.out, lrt3.out, lrt4.out, lrt5.out, lrt6.out, lrt7.out, lrt8.out, lrt9.out, lrt10.out, lrt11.out, lrt12.out, lrt13.out, lrt14.out, yy$counts))

head(lrtTot)

dim(lrtTot)

**# Ratio for heatmap**

all.samples<-as.data.frame(lcpmn)

#1. Controls,

hcf<-all.samples[,targets$Dx== "HC" & targets$Gender== "F" ]

hcfM<-rowMeans(hcf)

hcm<-all.samples[,targets$Dx== "HC" & targets$Gender== "M" ]

hcmM<-rowMeans(hcm)

hc<-all.samples[,targets$Dx== "HC" ]

hcM<-rowMeans(hc)

ibsfc<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "C" ]

ibsfcM<-rowMeans(ibsfc)

ibsfd<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "D" ]

ibsfdM<-rowMeans(ibsfd)

ibsfm<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "M" ]

ibsfmM<-rowMeans(ibsfm)

ibsf<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F"]

ibsfM<-rowMeans(ibsf)

#2. Exp,

ehcm<-all.samples[,targets$Dx== "HC" & targets$Gender== "M" ]

ibsmc<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "C" ]

ibsmd<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "D" ]

ibsmm<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "M" ]

ibsm<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" ]

ibsf<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" ]

ibs<-all.samples[,targets$Dx== "IBS" ]

ibsc<-all.samples[,targets$Dx== "IBS" & targets$BH\_Colon\_Exam== "C" ]

ibsd<-all.samples[,targets$Dx== "IBS" & targets$BH\_Colon\_Exam== "D" ]

ibsm<-all.samples[,targets$Dx== "IBS" & targets$BH\_Colon\_Exam== "M" ]

ibsmc<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "C" ]

ibsfc<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "C" ]

ibsmd<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "D" ]

ibsfd<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "D" ]

ibsmm<-all.samples[,targets$Dx== "IBS" & targets$Gender== "M" & targets$BH\_Colon\_Exam== "M" ]

ibsfm<-all.samples[,targets$Dx== "IBS" & targets$Gender== "F" & targets$BH\_Colon\_Exam== "M" ]

#3. Ratios

hc.mf<- ehcm - hcfM

colnames(hc.mf)<-paste(colnames(hc.mf), "\_vs\_HC.F.N" ,sep="")

ibsc.mf<- ibsmc - ibsfcM

colnames(ibsc.mf)<-paste(colnames(ibsc.mf), "\_vs\_IBS.F.C" ,sep="")

ibsd.mf<- ibsmd - ibsfdM

colnames(ibsd.mf)<-paste(colnames(ibsd.mf), "\_vs\_IBS.F.D" ,sep="")

ibsm.mf<- ibsmm - ibsfdM

colnames(ibsm.mf)<-paste(colnames(ibsm.mf), "\_vs\_IBS.F.M" ,sep="")

ibs.mf<- ibsm - ibsfM

colnames(ibs.mf)<-paste(colnames(ibs.mf), "\_vs\_IBS.F" ,sep="")

ibshc.m<- ibsm - hcmM

colnames(ibshc.m)<-paste(colnames(ibshc.m), "\_vs\_HC.M" ,sep="")

ibshc.f<- ibsf - hcfM

colnames(ibshc.f)<-paste(colnames(ibshc.f), "\_vs\_HC.F" ,sep="")

ibshc<- ibs - hcM

colnames(ibshc)<-paste(colnames(ibshc), "\_vs\_HC" ,sep="")

ibsmc.hcn<- ibsmc - hcM

colnames(ibsmc.hcn)<-paste(colnames(ibsmc.hcn), "\_vs\_HC.M.N" ,sep="")

ibsmd.hcn<- ibsmd - hcM

colnames(ibsmd.hcn)<-paste(colnames(ibsmd.hcn), "\_vs\_HC.M.N" ,sep="")

ibsmm.hcn<- ibsmm - hcM

colnames(ibsmm.hcn)<-paste(colnames(ibsmm.hcn), "\_vs\_HC.M.N" ,sep="")

ibsfc.hcn<- ibsfc - hcM

colnames(ibsfc.hcn)<-paste(colnames(ibsfc.hcn), "\_vs\_HC.F.N" ,sep="")

ibsfd.hcn<- ibsfd - hcM

colnames(ibsfd.hcn)<-paste(colnames(ibsfd.hcn), "\_vs\_HC.F.N" ,sep="")

ibsfm.hcn<- ibsfm - hcM

colnames(ibsfm.hcn)<-paste(colnames(ibsfm.hcn), "\_vs\_HC.F.N" ,sep="")

**##arrange output**

colnames(lcpmn)<-labls

colnames(lcpmn)<-paste(colnames(lcpmn), ".logCPM", sep="")

#exporting all the data

ratio.exp <- as.data.frame(cbind(yy$genes, hc.mf, ibsc.mf, ibsd.mf, ibsm.mf, ibs.mf, ibshc.m , ibshc.f, ibshc, ibsmc.hcn, ibsfc.hcn, ibsmd.hcn , ibsfd.hcn ,ibsmm.hcn , ibsfm.hcn , yy$counts, lcpmn ))

**#output complete gene list**

Annot <- y$genes

logFC <- lrtTot[ , grep("logFC", colnames(lrtTot))]

pVal <- lrtTot[ , grep("PValue", colnames(lrtTot))]

FDR <- lrtTot[ , grep("FDR", colnames(lrtTot))]

logCPM <- lrtTot[, grep("logCPM", colnames(lrtTot))]

LR <- lrtTot[ , grep("LR", colnames(lrtTot))]

Rank <- rank(rowSums(-normC), ties.method="first")

lrtAll<-as.data.frame(cbind(Annot, Rank,logFC,pVal,FDR,LR,logCPM))

Complete<-merge(lrtAll,ratio.exp,by.x="gene\_id",by.y="gene\_id")

write.csv(Complete, file= "Complete\_geneList.csv")

**##significant list**

Sig\_fdr0.05<- Complete[Complete$FDR\_HC.M.N\_vs\_HC.F.N <0.05 | Complete$FDR\_IBS.M.C\_vs\_IBS.F.C <0.05 |

Complete$FDR\_IBS.M.D\_vs\_IBS.F.D <0.05 |

Complete$FDR\_IBS.M.M\_vs\_IBS.F.M <0.05 |

Complete$FDR\_IBS.M\_vs\_IBS.F <0.05 |

Complete$FDR\_IBS.M\_vs\_HC.M <0.05 |

Complete$FDR\_IBS.F\_vs\_HC.F <0.05 |

Complete$FDR\_IBS\_vs\_HC <0.05 |

Complete$FDR\_IBS.M.C\_vs\_HC.M.N<0.05 |

Complete$FDR\_IBS.F.C\_vs\_HC.F.N<0.05 |

Complete$FDR\_IBS.M.D\_vs\_HC.M.N<0.05 |

Complete$FDR\_IBS.F.D\_vs\_HC.F.N<0.05 |

Complete$FDR\_IBS.M.M\_vs\_HC.M.N<0.05 |

Complete$FDR\_IBS.F.M\_vs\_HC.F.N<0.05,]

dim(Sig\_fdr0.05)

)

write.csv(Sig\_fdr0.05, file= "Significant\_geneList\_FDR0.05.csv")

Sig\_fdr0.1<- Complete[Complete$FDR\_HC.M.N\_vs\_HC.F.N <0.1 | Complete$FDR\_IBS.M.C\_vs\_IBS.F.C <0.1 |

Complete$FDR\_IBS.M.D\_vs\_IBS.F.D <0.1 |

Complete$FDR\_IBS.M.M\_vs\_IBS.F.M <0.1 |

Complete$FDR\_IBS.M\_vs\_IBS.F <0.1 |

Complete$FDR\_IBS.M\_vs\_HC.M <0.1 |

Complete$FDR\_IBS.F\_vs\_HC.F <0.1 |

Complete$FDR\_IBS\_vs\_HC <0.1 |

Complete$FDR\_IBS.M.C\_vs\_HC.M.N<0.1 |

Complete$FDR\_IBS.F.C\_vs\_HC.F.N<0.1 |

Complete$FDR\_IBS.M.D\_vs\_HC.M.N<0.1 |

Complete$FDR\_IBS.F.D\_vs\_HC.F.N<0.1 |

Complete$FDR\_IBS.M.M\_vs\_HC.M.N<0.1 |

Complete$FDR\_IBS.F.M\_vs\_HC.F.N<0.1,]

dim(Sig\_fdr0.1)

write.csv(Sig\_fdr0.1, file= "Significant\_geneList\_FDR0.1.csv")

Sig\_p0.005<- Complete[Complete$PValue\_HC.M.N\_vs\_HC.F.N <0.005 | Complete$PValue\_IBS.M.C\_vs\_IBS.F.C <0.005 |

Complete$PValue\_IBS.M.D\_vs\_IBS.F.D <0.005 |

Complete$PValue\_IBS.M.M\_vs\_IBS.F.M <0.005 |

Complete$PValue\_IBS.M\_vs\_IBS.F <0.005 |

Complete$PValue\_IBS.M\_vs\_HC.M <0.005 |

Complete$PValue\_IBS.F\_vs\_HC.F <0.005 |

Complete$PValue\_IBS\_vs\_HC <0.005 |

Complete$PValue\_IBS.M.C\_vs\_HC.M.N<0.005 |

Complete$PValue\_IBS.F.C\_vs\_HC.F.N<0.005 |

Complete$PValue\_IBS.M.D\_vs\_HC.M.N<0.005 |

Complete$PValue\_IBS.F.D\_vs\_HC.F.N<0.005 |

Complete$PValue\_IBS.M.M\_vs\_HC.M.N<0.005 |

Complete$PValue\_IBS.F.M\_vs\_HC.F.N<0.005,]

dim(Sig\_p0.005)

write.csv(Sig\_p0.005, file= "Significant\_geneList\_pLess0.005.csv")

**###plot heapmap with significant genes**

toplot5<-Sig\_p0.005[,66:442]

toplot5a<- toplot5 [,1:16]

toplot5b<- toplot5 [,17:51]

toplot5c<- toplot5 [,52:86]

toplot5d<- toplot5 [,87:183]

toplot5e<- toplot5 [,184:280]

toplot5g<- toplot5 [,281:311]

toplot5h<- toplot5 [,312:346]

toplot5j<- toplot5 [,347:377]

pdf("5\_heatmap\_p0.005.pdf", height=10, width=10)

par(mar=c(5,4,2.5,3))

heatmap.2(as.matrix(toplot5a), col=rev(redgreen(68)), main="HC, Male vs. Femal, p<0.005", trace="none", breaks=breaks, margins=c(6,5), cexCol=0.7, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5b), col=rev(redgreen(68)), main="IBS, Male vs. Female within each Exam, p<0.005", trace="none", breaks=breaks, margins=c(6,5), ColSideColors=colu1, cexCol=0.5, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5c), col=rev(redgreen(68)), main="IBS, Male vs. Female, p<0.005", trace="none", breaks=breaks, margins=c(6,5), cexCol=0.7, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5d), col=rev(redgreen(68)), main="IBS vs. HC, within each Gender, p<0.005",trace="none", breaks=breaks, margins=c(6,5), ColSideColors=colu2, cexCol=0.5, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5e), col=rev(redgreen(68)), main="IBS vs. HC, p<0.005", trace="none", breaks=breaks, margins=c(6,5), cexCol=0.4, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5g), col=rev(redgreen(68)), main="IBS ExamC vs. HC Female ExamN, p<0.005", trace="none", breaks=breaks, margins=c(6,5), ColSideColors=colu3, cexCol=0.7, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5h), col=rev(redgreen(68)), main="IBS ExamD vs. HC ExamN, p<0.005", trace="none", breaks=breaks, margins=c(6,5), cexCol=0.7, ColSideColors=colu4, cexRow=0.7, keysize=1, labRow = "")

heatmap.2(as.matrix(toplot5j), col=rev(redgreen(68)), main="IBS ExamM vs. HC ExamN, p<0.005", trace="none", breaks=breaks, margins=c(6,5), cexCol=0.7, ColSideColors=colu5, cexRow=0.7, keysize=1, labRow = "")

dev.off()

**##p-value distribution**

pdf("2-pVal\_histogram.pdf", height=8,width=10)

par(mfrow=c(4,4))

for(i in 1:ncol(pVal)){

hist(as.matrix(pVal[i]), breaks=100, col=rainbow(ncol(pVal))[i], main=names(pVal[i]))}

dev.off()

#

pdf("3\_FDR\_histogram.pdf", height=8,width=10)

par(mfrow=c(4,4))

for(i in 1:ncol(FDR)){

hist(as.matrix(FDR[i]), breaks=100, col=rainbow(ncol(FDR))[i], main=names(FDR[i]))}

dev.off()

**# barplot for number of sig genes**

pdf("1\_contrastAnalysis\_p0.005.pdf", height=10, width=12)

par(mar=c(3,3.5,2.5,3.5))

lfc <- Sig\_p0.005 [,c(7:34)]

head(lfc)

dim(lfc)

ups <- NA

downs <- NA

for(i in 1:ncol(FDR))

{

downs <- c(downs, -length(which(lfc[which(lfc[i+ncol(FDR)] <= 0.005),i] < 0)))

ups <- c(ups, length(which(lfc[which(lfc[i+ncol(FDR)] <= 0.005),i]>0)))

}

print(ups)

print(downs)

mx <- max(ups[is.na(ups)==FALSE])

mn <- min(downs[is.na(downs)==FALSE])

bp1 <- barplot(downs,horiz=TRUE,xlim=c(mn,mx), col="green", )

bp2 <- barplot(ups, horiz=TRUE,xlim=c(mn,mx), col="red",add=TRUE,axes=TRUE)

axis(2, at=bp1[2:length(bp1)],tick=FALSE,labels=downs[is.na(downs)==FALSE],las=1)

axis(4,at=bp2[2:length(bp2)],tick=FALSE,labels=ups[is.na(ups)==FALSE], las=1)

#axis(4,at=bp2[2:length(bp2)],tick=FALSE,labels=ups[is.na(ups)==FALSE], las=1,line=-2)

labs <- NA

fullNames <- gsub("logFC\_" ,"", colnames(logFC[,1:ncol(logFC)])) # DO THIS RIGHT ORDER!!!

#fullNames <- colnames(logFC[,1:ncol(logFC)])

for(tis in fullNames){

labs <- c(labs, tis)

}

labs <- labs[is.na(labs)==FALSE]

text(x=0,y=bp2[2:length(bp2)], labels=labs)

title(main=paste("Gene Changes, p <",padj, "(",dim(SigGene05)[[1]], ")", sep=" "))

dev.off()

**#R sessionInfo()**

R version 3.4.2 (2017-09-28)

Platform: x86\_64-w64-mingw32/x64 (64-bit)

Running under: Windows 7 x64 (build 7601) Service Pack 1

Matrix products: default

locale:

[1] LC\_COLLATE=English\_United States.1252 LC\_CTYPE=English\_United States.1252

[3] LC\_MONETARY=English\_United States.1252 LC\_NUMERIC=C

[5] LC\_TIME=English\_United States.1252

attached base packages:

[1] parallel stats graphics grDevices utils datasets methods base

other attached packages:

[1] RColorBrewer\_1.1-2 DESeq\_1.28.0 lattice\_0.20-35 locfit\_1.5-9.1

[5] Biobase\_2.36.2 BiocGenerics\_0.22.1 edgeR\_3.18.1 limma\_3.32.10

[9] MASS\_7.3-47 heatmap.plus\_1.3 gplots\_3.0.1

loaded via a namespace (and not attached):

[1] Rcpp\_0.12.14 compiler\_3.4.2 pillar\_1.0.1

[4] bitops\_1.0-6 digest\_0.6.13 bit\_1.1-12

[7] annotate\_1.54.0 RSQLite\_2.0 memoise\_1.1.0

[10] tibble\_1.4.1 rlang\_0.1.6 Matrix\_1.2-11

[13] DBI\_0.7 genefilter\_1.58.1 S4Vectors\_0.14.7

[16] gtools\_3.5.0 caTools\_1.17.1 IRanges\_2.10.5

[19] stats4\_3.4.2 bit64\_0.9-7 grid\_3.4.2

[22] AnnotationDbi\_1.38.2 XML\_3.98-1.9 survival\_2.41-3

[25] gdata\_2.18.0 geneplotter\_1.54.0 blob\_1.1.0

[28] splines\_3.4.2 xtable\_1.8-2 KernSmooth\_2.23-15

[31] RCurl\_1.95-4.10