Figure 1: Folder arrangements in the example.

Figure 2: file\_disease\_state.csv

##LOAD oligo LBRARY

library("oligo") ##FOR HuGene S.T 1.0 array

###########################################

# READING .CEL FILES #

###########################################

setwd("E:/labWork\_SGPGI/NAFLD/Data\_collection/Study-4/analysis")

#file\_disease\_sate.csv (IN analysis FOLDER): CONTAINS [" ",celFile,age,gender,diseaseState] COLUMNS (SEE FIGURE 2)

files=read.table(file ="file\_disease\_state.csv", header = TRUE, sep = ",")

#EXTRACT THE .CEL FILE NAMES FROM 'file\_disease\_sate.csv' INTO A VECTOR

CELfiles=as.character(files$celFile)

head(CELfiles) ##[1] "16.CEL" "17.CEL" "18.CEL" "21.CEL" "22.CEL" "47.CEL"

#GO TO THE FOLDER CONTAINING .CEL FILES

setwd("E:/labWork\_SGPGI/NAFLD/Data\_collection/Study-4/cel\_files")

#READ REQUIRED .CEL FILES (ARRAY INDEX)INTO A 'GeneFeatureSet' OBJECT NAMED 'rawdata.normal\_st'

rawdata.normal\_st=read.celfiles( filenames= c(CELfiles[16:34],CELfiles[35:43]))

#COME BACK TO 'analysis' FOLDER

setwd("E:/labWork\_SGPGI/NAFLD/Data\_collection/Study-4/analysis")

##READING .CEL FILES \*\*DONE\*\*

###########################################

# PROCESSING RAW INTESITY DATA #

###########################################

# Background correction, normalization and Calculating Expression summarization using RMA

# the normdatas are objects of ExpressionSet class.

normdata.normal\_st=rma(rawdata.normal\_st)

#################################################

# QUALITY CONTRLS #

#################################################

##BOX-PLOTS FOR RAW AND NORMALIZED INTENSITIES

#EXTRACT EXPRESSION VALUES FROM RAW DATA AND ADD CONDITION LABELS (e.g Normal AND Steatosis)

edat.normal\_st.raw=exprs(rawdata.normal\_st)

colnames(edat.normal\_st.raw)=c(paste("Normal",1:19), paste("Steatosis", 1:9))

#EXTRACT EXPRESSION VALUES FROM NORMALIZED DATA AND ADD CONDITION LABELS

edat.normal\_st.norm=exprs(normdata.normal\_st)

colnames(edat.normal\_st.norm)=c(paste("Normal",1:19), paste("Steatosis", 1:9))

##EXPORT AND SAVE NORMALIZED DATA INTO FILES

write.table(edat.normal\_st.norm, file="edat.normal\_st.norm.annot.xls", sep="\t", col.names = NA)

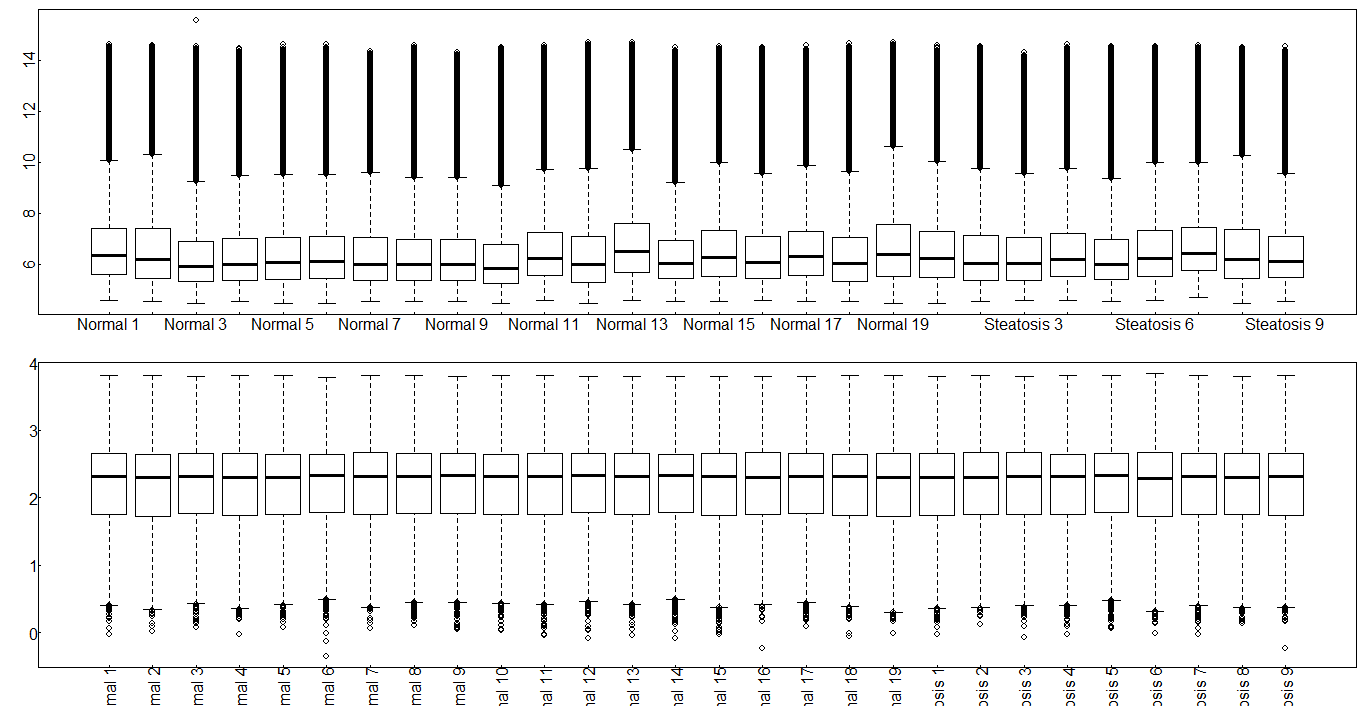
##GRAPHICAL PARAMETER SETTING

par(mfrow = c(2,1), mar=c(2,2,0.5,0.5), mgp = c(0.7,0, 0), tck=0.008)

##PLOT log2 EXPRESSION VALUES

boxplot(log2(edat.normal\_st.raw),cex=0.5, las=2)

boxplot(log2(edat.normal\_st.norm),cex=0.5, las=2)



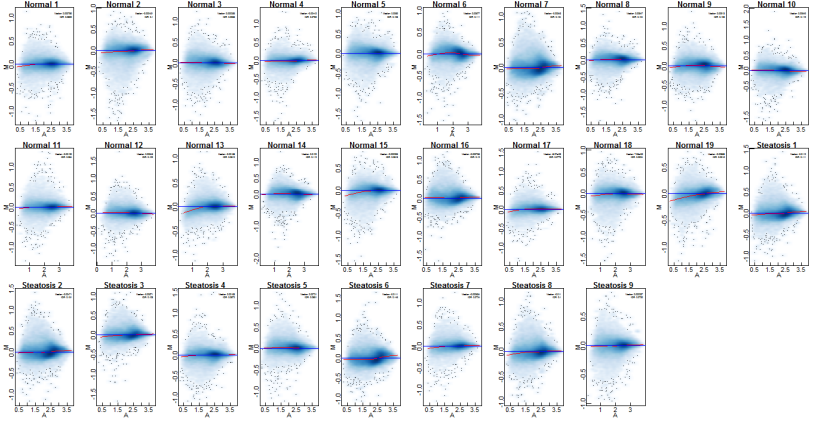
##MAplot raw expresssion data

##GRAPHICAL PARAMETER SETTING

par(mfrow = c(3,10), mar=c(2,2,0.5,0.5), mgp = c(0.7,0, 0), tck=0.08)

#MAplot FOR RAW VALUES

MAplot(log2(edat.normal\_st.raw),cex=0.4, main=" ")



par(mfrow = c(3,10), mar=c(2,2,0.5,0.5), mgp = c(0.7,0, 0), tck=0.08)

MAplot(log2(edat.normal\_st.norm),cex=0.4, main=" ")

#################################################

##DIFFEENTIAL GENE EXPRESSION identification #

## USING limma PACKAGE #

#################################################

library("limma")

### DESIGN MATRIX AND CONTRAST MATRIX

######################################

#The package limma uses an approach called linear models to analyze designed #microarray experiments. This approach allows very general experiments to be #analyzed just as easily as a simple replicated experiment.

#The approach requires one or two matrices to be specified.

# 1.design matrix :which indicates in effect which RNA samples have been applied to each array.

# 2.contrast matrix :which specifies which comparisons you would like to make #between the RNA samples. For very simple experiments, you may not need to specify #the contrast matrix.

#Each row of the design matrix corresponds to an array in your experiment

#and each column corresponds to a coefficient which is used to describe the RNA #sources in your experiment. With Affymetrix you will need as many coefficients as #you have distinct RNA sources, no more and no less.

##WE HAVE TWO GROUPS NAMELY 'Normal' AND 'Steatosis'

##Normal=1 AND Steatosis=2; Normal HAS 19 CEL FILES AND Steatosis HAS 9 CEL FILES

design <- model.matrix(~ 0+factor(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,2)))

#ADD COLUMN NAMES

colnames(design) <- c("Normal", "steatosis")

*#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*> design*

*Normal steatosis*

*1 1 0*

*2 1 0*

*3 1 0*

*4 1 0*

*5 1 0*

*6 1 0*

*. . .*

*. . .*

*. . .*

*24 0 1*

*25 0 1*

*26 0 1*

*27 0 1*

*28 0 1*

*###########################*

##IN CONTRAST MATRIX WE HAVE TO SPECIFY WHAT PAIR WE WANT TO COMPARE,

cont.matrix <- makeContrasts(STvsN=steatosis-Normal, levels=design)

*#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*>cont.matrix*

*Contrasts*

*Levels STvsN*

*Normal -1*

*steatosis 1*

*#################################*

##CREATE MArrayLM OBJECTS BY lmFit AND eBayes

fit <- lmFit(edat.normal\_st.norm, design)

fit2 <- contrasts.fit(fit, cont.matrix)

fit2 <- eBayes(fit2)

##FIND AND FILTER THE OPGENES IN THE MATRIX ON DIFFERENT PARAMETERS

#coef= column number or column name specifying which coefficient or #contrast of the linear model is of interest.

#adjust.method="none", "BH", "BY" and "holm"

#resort.by="logFC", "AveExpr", "t", "P", "p", "B" or "none"

#lfc=minimum absolute log2-fold-change required

TopGenes\_all=topTable(fit2,number=Inf, adjust.method="BH",sort.by="B", p.value=1, lfc=0)

#################################################

# ANNOTATION TO TopGenes #

#################################################

##NEED TO INSTALL PROBESET ANNOTATION LIBRARY (e.g hugene10sttranscriptcluster.db)

library(annotate)

library("hugene10sttranscriptcluster.db")

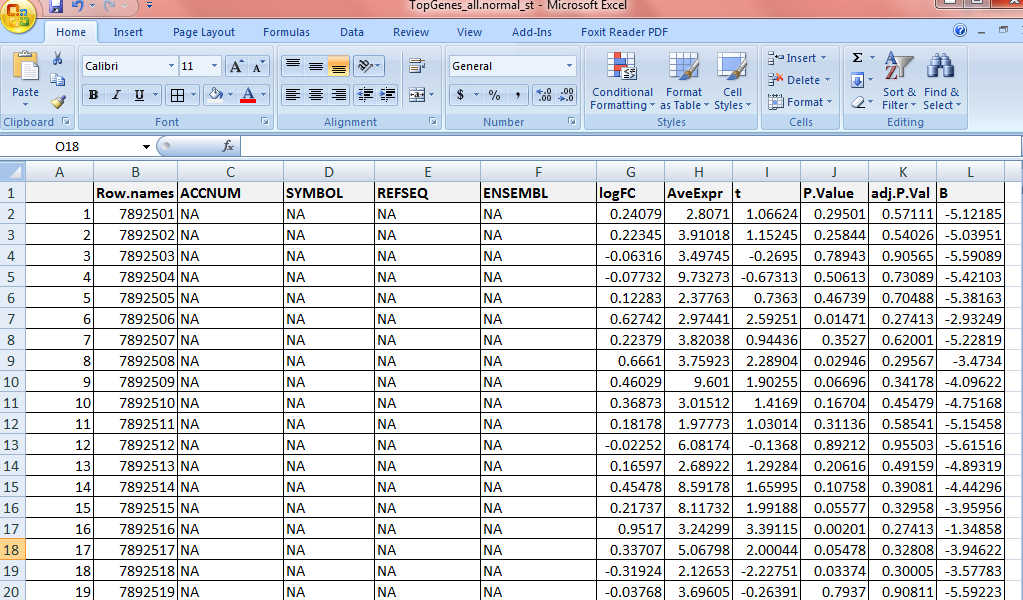
##ADD AS MANY ANNOTATIONS AS YOU CAN, SEE hugene10sttranscriptcluster.db MANNUAL FOR DATAILS.

Annot= data.frame(ACCNUM=sapply(contents(hugene10sttranscriptclusterACCNUM),paste, collapse=","),SYMBOL=sapply(contents(hugene10sttranscriptclusterSYMBOL),paste, collapse=","),REFSEQ=sapply(contents(hugene10sttranscriptclusterREFSEQ),paste,collapse=","),ENSEMBL=sapply(contents(hugene10sttranscriptclusterENSEMBL),paste, collapse=","))

TopGenes\_all.annot=merge(Annot,TopGenes\_all,by.x=0, by.y=0)

##SAVE RESULTS TO FILE

write.table(TopGenes\_all.annot, file="TopGenes\_all.normal\_st.xls", sep="\t", col.names = NA)



sessionInfo()

R version 3.1.2 (2014-10-31)

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

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] oligoClasses\_1.28.0 Biobase\_2.26.0 BiocGenerics\_0.12.1 limma\_3.22.1

loaded via a namespace (and not attached):

[1] affyio\_1.34.0 BiocInstaller\_1.16.1 Biostrings\_2.34.1 bit\_1.1-12

[5] codetools\_0.2-9 ff\_2.2-13 foreach\_1.4.2 GenomeInfoDb\_1.2.4

[9] GenomicRanges\_1.18.3 IRanges\_2.0.1 iterators\_1.0.7 S4Vectors\_0.4.0

[13] stats4\_3.1.2 tools\_3.1.2 XVector\_0.6.0 zlibbioc\_1.12.0