########identifying FA-altered genus in cultivar D72#############

library(DESeq2); library(dplyr); library(reshape2); library (ggplot2); library(ggpubr); library (ggrepel)

design\_filter\_16s<-read.table("design\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

tax\_filter\_16s<- read.table("tax\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

otu\_filter\_16s<-read.table("otu\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

otu\_filter\_16s <- as.matrix(otu\_filter\_16s) #rownames (otu\_filter\_16s); #colnames (otu\_filter\_16s)

dim(otu\_filter\_16s)

#####Genus abundance#####

myBtaxa <- tax\_filter\_16s

otu\_its\_RA <- otu\_filter\_16s

myBtaxa$Genus.full <- paste(myBtaxa$Phylum, myBtaxa$Class, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Order, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Family, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Genus, sep=",")

myBtaxa$ OTUnames <- rownames(myBtaxa)

not.unass <- rownames(myBtaxa)[ myBtaxa$Genus!="unassigned" ]

for(i in not.unass){

myBtaxa [rownames(myBtaxa)[ myBtaxa $ OTUnames ==paste(i)], ]$Genus.full <- myBtaxa [rownames(myBtaxa)==paste(i),]$Genus }

PHYLAnames\_its <- names(sort(table(myBtaxa[,"Genus.full"]), decr=T))

length(PHYLAnames\_its)

## Preparation of matrix with relative abundance by class

y <- NULL

otunames <- rownames(otu\_its\_RA)

for (i in PHYLAnames\_its){

x <- array(colSums(otu\_its\_RA [rownames(myBtaxa)[which(myBtaxa$Genus.full == paste(i))],,drop=FALSE]))

y <- rbind(y,x) }

## Create matrix

rownames(y) <- paste(PHYLAnames\_its)

colnames(y) <- paste(colnames(otu\_its\_RA))

CLASS\_mat\_its <- y

colSums(CLASS\_mat\_its)

CLASS\_mat\_its\_mean <- sort(apply(CLASS\_mat\_its,1,mean),decr=T)

CLASS\_mat\_its <- CLASS\_mat\_its[names(CLASS\_mat\_its\_mean),]

dim(CLASS\_mat\_its)

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

design <- design\_filter\_16s; otu.data <- CLASS\_mat\_its

library (dplyr)

design <- **filter**(design, Cultivar=="D72")

design.FC <- design

otu.data.FC <- otu.data[, rownames(design.FC)]

otu.data.FC <- otu.data.FC [rowSums(otu.data.FC)>0, ]

Pct <- 0.01; BM=50; CK.rep=4; Trt.rep=4

otu.data <- as.data.frame(otu.data.FC)

Mydesign <- data.frame(Trt = factor(c(rep(c('control'), CK.rep), rep(c('treat'), Trt.rep)), levels = c('control', 'treat')))

dds <- DESeqDataSetFromMatrix( countData = otu.data, colData = Mydesign , design= ~Trt )

dds1 <- DESeq(dds, fitType = 'mean', minReplicatesForReplace = 7, parallel = F)

res <- results(dds1, contrast = c('Trt', 'treat', 'control'))

res1 <- data.frame(res, stringsAsFactors = FALSE, check.names = FALSE)

###################MA plots##########

tt\_enrich <- as.data.frame(res1)

tt\_enrich$Genus <- rownames(tt\_enrich)

forMA <- tt\_enrich

forMA <- mutate(forMA, signif = ifelse(padj <= Pct, "T", "F"))

C\_enrich <- filter(forMA, log2FoldChange > 0 & baseMean>BM & padj< Pct) ; dim (C\_enrich) # Enriched

CK\_enrich <- filter(forMA, log2FoldChange < 0 & baseMean>BM & padj< Pct) ; dim (CK\_enrich) # Depleted

cs <- c(rownames(CK\_enrich), rownames(C\_enrich) )

cs.T.En <- rownames(C\_enrich); cs.T.De <- rownames(CK\_enrich)

## define colors

forMA$col <- "dimgrey"

forMA [which(rownames(forMA) %in% rownames(CK\_enrich) ), ]$col <- "darkorange"

forMA [which(rownames(forMA) %in% rownames(C\_enrich) ), ]$col <- "dodgerblue"

# order for plotting colors

forMA$ord[forMA$col=="dimgrey"] <- 1; forMA$ord[forMA$col=="darkorange"] <- 2; forMA$ord[forMA$col=="dodgerblue"] <- 3

forMA <- forMA[sort(forMA$ord, ind=T, decr=F) $ ix, ]

## define pch

forMA$pch <- 1 ; forMA [which(rownames(forMA) %in% cs ), ] $pch <- 16

## define size

forMA$size <- 1; forMA [which(rownames(forMA) %in% cs ), ]$size <- 1.5

forMA$label <- NA

forMA $ OTUnames <- rownames(forMA)

for(i in cs){ forMA [rownames(forMA)[ forMA $ OTUnames ==paste(i)], ]$label <- forMA [rownames(forMA)==paste(i),]$Genus}

**write.table**(forMA, 'forMA.D72.Genus.FA.txt', col.names = NA, sep = '\t', quote = FALSE)

xrng <- range(log(forMA$baseMean)) ; yrng <- range(forMA$log2FoldChange)

MA.Sys.D72 <- ggplot(data=forMA, aes(x= log(baseMean), y= log2FoldChange, colour=col))+

geom\_point(aes(shape= factor(pch), size=size) )+

scale\_color\_manual(values=c("forestgreen", "dimgrey", "dodgerblue"), guide="none" )+

scale\_shape\_manual(values = c(1, 16) , guide="none")+

scale\_size\_continuous(range = c(0.9, 1.4), guide="none")+

labs(x="Average abundance (log(BM))", y="log2 fold change")+

**geom\_text\_repel**(data=forMA, aes(x=log(baseMean), y= log2FoldChange, label=label), colour="black", size=3) +

annotate("text",label=paste("CK enriched: ", format(length(rownames(CK\_enrich))), sep=""), color="forestgreen",

x=xrng[2]-abs(xrng[1])\*0.5, y=yrng[1], size=4, vjust="inward", hjust="inward")+

annotate("text", label=paste("C enriched: ", format(length(rownames(C\_enrich))), sep=""), color= "dodgerblue",

x=xrng[2]-abs(xrng[1])\*0.5, y=yrng[2], size=4, vjust="inward", hjust="inward")+

theme\_bw()+

theme(panel.grid=element\_blank(),

axis.text=element\_text(size=10, color="black"), axis.title=element\_text(size=12),

legend.background=element\_rect(fill=NA, linewidth=0.6, linetype="dashed", colour ="grey20"),

legend.text=element\_text(size=12) )

MA.Sys.D72

########identifying FA-altered genus in cultivar Z19#############

library(DESeq2); library(dplyr); library(reshape2); library (ggplot2); library(ggpubr); library (ggrepel)

design\_filter\_16s<-read.table("design\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

tax\_filter\_16s<- read.table("tax\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

otu\_filter\_16s<-read.table("otu\_filter\_16s.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

otu\_filter\_16s <- as.matrix(otu\_filter\_16s) #rownames (otu\_filter\_16s); #colnames (otu\_filter\_16s)

dim(otu\_filter\_16s)

#####Genus abundance#####

myBtaxa <- tax\_filter\_16s

otu\_its\_RA <- otu\_filter\_16s

myBtaxa$Genus.full <- paste(myBtaxa$Phylum, myBtaxa$Class, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Order, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Family, sep=",")

myBtaxa$Genus.full <- paste(myBtaxa$Genus.full, myBtaxa$Genus, sep=",")

myBtaxa$ OTUnames <- rownames(myBtaxa)

not.unass <- rownames(myBtaxa)[ myBtaxa$Genus!="unassigned" ]

for(i in not.unass){

myBtaxa [rownames(myBtaxa)[ myBtaxa $ OTUnames ==paste(i)], ]$Genus.full <- myBtaxa [rownames(myBtaxa)==paste(i),]$Genus }

PHYLAnames\_its <- names(sort(table(myBtaxa[,"Genus.full"]), decr=T))

length(PHYLAnames\_its)

## Preparation of matrix with relative abundance by class

y <- NULL

otunames <- rownames(otu\_its\_RA)

for (i in PHYLAnames\_its){

x <- array(colSums(otu\_its\_RA [rownames(myBtaxa)[which(myBtaxa$Genus.full == paste(i))],,drop=FALSE]))

y <- rbind(y,x) }

## Create matrix

rownames(y) <- paste(PHYLAnames\_its)

colnames(y) <- paste(colnames(otu\_its\_RA))

CLASS\_mat\_its <- y

colSums(CLASS\_mat\_its)

CLASS\_mat\_its\_mean <- sort(apply(CLASS\_mat\_its,1,mean),decr=T)

CLASS\_mat\_its <- CLASS\_mat\_its[names(CLASS\_mat\_its\_mean),]

dim(CLASS\_mat\_its)

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

design <- design\_filter\_16s; otu.data <- CLASS\_mat\_its

library (dplyr)

design <- **filter**(design, Cultivar=="Z19")

design.FC <- design

otu.data.FC <- otu.data[, rownames(design.FC)]

otu.data.FC <- otu.data.FC [rowSums(otu.data.FC)>0, ]

Pct <- 0.01; BM=50; CK.rep=4; Trt.rep=4

otu.data <- as.data.frame(otu.data.FC) ; CK.rep=4; Trt.rep=4

Mydesign <- data.frame(Trt = factor(c(rep(c('control'), CK.rep), rep(c('treat'), Trt.rep)), levels = c('control', 'treat')))

dds <- DESeqDataSetFromMatrix( countData = otu.data, colData = Mydesign , design= ~Trt )

dds1 <- DESeq(dds, fitType = 'mean', minReplicatesForReplace = 7, parallel = F)

res <- results(dds1, contrast = c('Trt', 'treat', 'control'))

res1 <- data.frame(res, stringsAsFactors = FALSE, check.names = FALSE)

###################MA plots##########

tt\_enrich <- as.data.frame(res1)

tt\_enrich$Genus <- rownames(tt\_enrich)

forMA <- tt\_enrich

forMA <- mutate(forMA, signif = ifelse(padj <= Pct, "T", "F"))

C\_enrich <- filter(forMA, log2FoldChange > 0 & baseMean>BM & padj<Pct) ; dim (C\_enrich) # Enriched

CK\_enrich <- filter(forMA, log2FoldChange < 0 & baseMean>BM & padj<Pct) ; dim (CK\_enrich) # Depleted

cs <- c(rownames(CK\_enrich), rownames(C\_enrich) )

cs.T.En <- rownames(C\_enrich); cs.T.De <- rownames(CK\_enrich)

## define colors

forMA$col <- "dimgrey"

forMA [which(rownames(forMA) %in% rownames(CK\_enrich) ), ]$col <- "darkorange"

forMA [which(rownames(forMA) %in% rownames(C\_enrich) ), ]$col <- "dodgerblue"

# order for plotting colors

forMA$ord[forMA$col=="dimgrey"] <- 1; forMA$ord[forMA$col=="darkorange"] <- 2; forMA$ord[forMA$col=="dodgerblue"] <- 3

forMA <- forMA[sort(forMA$ord, ind=T, decr=F) $ ix, ]

## define pch

forMA$pch <- 1 ; forMA [which(rownames(forMA) %in% cs ), ] $pch <- 16

## define size

forMA$size <- 1; forMA [which(rownames(forMA) %in% cs ), ]$size <- 1.5

forMA$label <- NA

forMA $ OTUnames <- rownames(forMA)

for(i in cs){ forMA [rownames(forMA)[ forMA $ OTUnames ==paste(i)], ]$label <- forMA [rownames(forMA)==paste(i),]$Genus}

**write.table**(forMA, 'forMA.Z19.Genus.FA.txt', col.names = NA, sep = '\t', quote = FALSE)

xrng <- range(log(forMA$baseMean)) ; yrng <- range(forMA$log2FoldChange)

MA.Sys.ZZ <- ggplot(data=forMA, aes(x= log(baseMean), y= log2FoldChange, colour=col))+

geom\_point(aes(shape= factor(pch), size=size) )+

scale\_color\_manual(values=c("forestgreen", "dimgrey", "dodgerblue"), guide="none" )+

scale\_shape\_manual(values = c(1, 16) , guide="none")+

scale\_size\_continuous(range = c(0.9, 1.4), guide="none")+

labs(x="Average abundance (log(BM))", y="log2 fold change")+

**geom\_text\_repel**(data=forMA, aes(x=log(baseMean), y= log2FoldChange, label=label), colour="black", size=3) +

annotate("text",label=paste("CK enriched: ", format(length(rownames(CK\_enrich))), sep=""), color="forestgreen",

x=xrng[2]-abs(xrng[1])\*0.5, y=yrng[1], size=4, vjust="inward", hjust="inward")+

annotate("text", label=paste("C enriched: ", format(length(rownames(C\_enrich))), sep=""), color= "dodgerblue",

x=xrng[2]-abs(xrng[1])\*0.5, y=yrng[2], size=4, vjust="inward", hjust="inward")+

theme\_bw()+

theme(panel.grid=element\_blank(),

axis.text=element\_text(size=10, color="black"), axis.title=element\_text(size=12),

legend.background=element\_rect(fill=NA, linewidth=0.6, linetype="dashed", colour ="grey20"),

legend.text=element\_text(size=12) )

MA.Sys.ZZ