########identifying FA-altered OTUs 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)

design <- design\_filter\_16s; otu.data <- otu\_filter\_16s

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$Genus <- tax\_filter\_16s [rownames(tt\_enrich), ]$Genus

forMA <- tt\_enrich

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

D72.Sys\_En <- filter(forMA, log2FoldChange>0&baseMean>BM & padj< Pct); dim (D72.Sys\_En) # Enriched &baseMean>5

D72.Sys\_De <- filter(forMA, log2FoldChange<0&baseMean>BM & padj< Pct); dim (D72.Sys\_De) # Depleted &baseMean>5

cs <- c(rownames(D72.Sys\_En), rownames(D72.Sys\_De) )

## define colors

forMA$col <- "dimgrey"

forMA [which(rownames(forMA) %in% rownames(D72.Sys\_De) ), ]$col <- "darkorange"

forMA [which(rownames(forMA) %in% rownames(D72.Sys\_En) ), ]$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}

forMA [rownames(forMA)[forMA$Genus=="unassigned" & rownames(forMA) %in% cs ], ]$label <- NA

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

xrng <- range(log2(forMA$baseMean)) ; yrng <- range(forMA$log2FoldChange)

MA.D72.Sys <- ggplot(data=forMA, aes(x= log2(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(0.6, 1.2), guide="none")+

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

**geom\_text\_repel**(data=forMA, aes(x=log2(baseMean), y=log2FoldChange, label=label), colour="black", size=3) +

annotate("text",label=paste("Sys depleted: ", format(length(rownames(D72.Sys\_De))), sep=""), color= "forestgreen",

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

annotate("text", label=paste("Sys enriched: ", format(length(rownames(D72.Sys\_En))), 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.D72.Sys

########identifying FA-altered OTUs 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)

design <- design\_filter\_16s; otu.data <- otu\_filter\_16s

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)

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$Genus <- tax\_filter\_16s [rownames(tt\_enrich), ]$Genus

forMA <- tt\_enrich

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

Z19.Sys\_En <- filter(forMA, log2FoldChange>0 &baseMean>BM & padj< Pct) ; dim (Z19.Sys\_En) # Enriched &baseMean>5

Z19.Sys\_De <- filter(forMA, log2FoldChange<0 &baseMean>BM & padj< Pct) ; dim (Z19.Sys\_De) # Depleted &baseMean>5

cs <- c(rownames(Z19.Sys\_En), rownames(Z19.Sys\_De) )

## define colors

forMA$col <- "dimgrey"

forMA [which(rownames(forMA) %in% rownames(Z19.Sys\_De) ), ]$col <- "darkorange"

forMA [which(rownames(forMA) %in% rownames(Z19.Sys\_En) ), ]$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}

forMA [rownames(forMA)[forMA$Genus=="unassigned" & rownames(forMA) %in% cs ], ]$label <- NA

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

xrng <- range(log2(forMA$baseMean)) ; yrng <- range(forMA$log2FoldChange)

MA.Z19.Sys <- ggplot(data=forMA, aes(x= log2(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(0.6, 1.2), guide="none")+ #点的大小

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

**geom\_text\_repel**(data=forMA, aes(x=log2(baseMean), y=log2FoldChange, label=label), colour="black", size=3) +

annotate("text",label=paste("Sys depleted: ", format(length(rownames(Z19.Sys\_De))), sep=""), color= "forestgreen",

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

annotate("text", label=paste("Sys enriched: ", format(length(rownames(Z19.Sys\_En))), 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.Z19.Sys

#####Plotting log2FC values for FA-altered OTUs ######

library (ggplot2); library(ggpubr); library (ggrepel)

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

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

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

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

forMA.D72<-read.table("forMA.D72.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

forMA.Z19<-read.table("forMA.Z19.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

Pct <- 0.01; BM=50

library(dplyr)

D72.Sys\_En <- filter(forMA.D72, log2FoldChange>0&baseMean>BM & padj< Pct); dim (D72.Sys\_En)

D72.Sys\_De <- filter(forMA.D72, log2FoldChange<0&baseMean>BM & padj< Pct); dim (D72.Sys\_De)

Z19.Sys\_En <- filter(forMA.Z19, log2FoldChange>0&baseMean>BM & padj< Pct); dim (Z19.Sys\_En)

Z19.Sys\_De <- filter(forMA.Z19, log2FoldChange<0&baseMean>BM & padj< Pct); dim (Z19.Sys\_De)

Sys\_En = union( rownames(D72.Sys\_En), rownames(Z19.Sys\_En) )

Sys\_De = union( rownames(D72.Sys\_De), rownames(Z19.Sys\_De) )

De.En = intersect(Sys\_En, Sys\_De) # tax\_filter\_16s [intersect(Sys\_En, Sys\_De), ]

De.En.All = union(Sys\_En, Sys\_De)

Group.En = setdiff(Sys\_En, De.En) # tax\_filter\_16s [Group.En, ]

Group.De = setdiff(Sys\_De, De.En)

csOTUs.tax <- tax\_rarefy\_16s [De.En.All, ]

csOTUs.tax <-csOTUs.tax [order(csOTUs.tax[,2], csOTUs.tax[,3], csOTUs.tax[,6]), ]

csOTUs <- rownames (csOTUs.tax)

otu\_16s\_RA <- t(t(otu\_rarefy\_16s)/colSums(otu\_rarefy\_16s))\*100; colSums(otu\_16s\_RA)

otu\_RA <- as.data.frame(t(otu\_16s\_RA [csOTUs, ])); range(otu\_RA)

OTU.RA <- as.data.frame(otu\_rarefy\_16s)

OTU.RA$RA <- rowSums(otu\_rarefy\_16s)/sum(otu\_rarefy\_16s) \*100

csOTUs.dat <- csOTUs.tax

csOTUs.dat$RA <- 0

for (i in rownames(csOTUs.dat) ) { csOTUs.dat [rownames(csOTUs.dat)==paste(i),]$RA <- OTU.RA [rownames(OTU.RA)==paste(i),]$RA }

range(csOTUs.dat $RA)

csOTUs.dat$D72 <- 0

for (i in rownames(csOTUs.dat)) { csOTUs.dat [rownames(csOTUs.dat)==paste(i),]$D72 <- forMA.D72 [rownames(forMA.D72)==paste(i),]$log2FoldChange }

csOTUs.dat$Z19 <- 0

for (i in rownames(csOTUs.dat)) { csOTUs.dat [rownames(csOTUs.dat)==paste(i),]$Z19 <- forMA.Z19 [rownames(forMA.Z19)==paste(i),]$log2FoldChange }

csOTUs.dat$D72.p <- 0

for (i in rownames(csOTUs.dat)) { csOTUs.dat [rownames(csOTUs.dat)==paste(i),]$D72.p <- forMA.D72 [rownames(forMA.D72)==paste(i),]$padj }

csOTUs.dat$Z19.p <- 0

for (i in rownames(csOTUs.dat)) { csOTUs.dat [rownames(csOTUs.dat)==paste(i),]$Z19.p <- forMA.Z19 [rownames(forMA.Z19)==paste(i),]$padj }

csOTUs.dat [is.na(csOTUs.dat)] <- 1

library(dplyr)

csOTUs.dat <- **mutate**(csOTUs.dat, D72.p = ifelse(D72.p <= 0.01, "S", "NS"))

csOTUs.dat <- **mutate**(csOTUs.dat, Z19.p = ifelse(Z19.p <= 0.01, "S", "NS"))

csOTUs.dat$P <- paste(csOTUs.dat$D72.p, csOTUs.dat$Z19.p, sep="-")

csOTUs.dat$AA <- rownames(csOTUs.dat)

range(csOTUs.dat$RA)

FA.otu = ggplot(data= csOTUs.dat , aes(x= Z19, y= D72 ))+

geom\_point(aes( size= RA, color= P ) ) +

**geom\_text\_repel**(data= csOTUs.dat, aes(x= Z19 , y= D72, label=Genus), colour="black", size=3) +

scale\_size\_continuous(name="RA (%)", range = c(0, 6), breaks=c(0.05, 0.1, 0.5, 1, 2.5, 5)) +

scale\_color\_manual(values= c("yellowgreen", "turquoise3", "dodgerblue")) + #, guide="none"

geom\_hline(yintercept=0, linetype=3, linewidth=1, color='gray')+

geom\_vline(xintercept=0, linetype=3, linewidth=1, color='gray')+

theme\_bw()+

theme(panel.grid=element\_blank(), plot.title = element\_text(face="bold", hjust = 0.5),

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

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

legend.key = element\_blank(), legend.background= element\_blank() )

FA.otu