##### Import Data #####

rm(list=ls()); library (stringr); library (vegan);library (dplyr)

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

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

mydesign <- read.table("mydesign.M.txt", header=T, row.names=1, stringsAsFactors=F, na.strings="NA")

Sam.Rep=4

# create separate taxonomy label specifying classes of Proteobacteria

myBtaxa$labels <- myBtaxa$Phylum

myBtaxa[ rownames(myBtaxa)[myBtaxa$Class=="Alphaproteobacteria" ], ]$labels <- "Alphaproteobacteria"

myBtaxa[ rownames(myBtaxa)[myBtaxa $Class=="Betaproteobacteria" ], ]$labels <- "Betaproteobacteria"

myBtaxa[ rownames(myBtaxa)[myBtaxa $Class=="Gammaproteobacteria" ], ]$labels <- "Gammaproteobacteria"

myBtaxa[ rownames(myBtaxa)[myBtaxa $Class=="Deltaproteobacteria" ], ]$labels <- "Deltaproteobacteria"

##### Store Archaea, Cyanobacteria and mitrochrondia sequeces #####

#unique(myBtaxa$Kingdom)

#table(myBtaxa$Kingdom)

r1 <- rownames(myBtaxa[myBtaxa$Kingdom=="Archaea",]); r1

#table(myBtaxa$Phylum)

r2 <- rownames(myBtaxa[myBtaxa$Phylum=="Cyanobacteria",]); r2

##unique(myBtaxa$Family)

r3 <- rownames(myBtaxa[myBtaxa$Family=="Mitochondria",]); r3

#otus\_remove\_16s <- c(r1,r2)

otus\_remove\_16s <- c(r1, r2, r3)

## Remove these from otu table, tax table

otu\_filter\_16s <- myBotu[-which(rownames(myBotu) %in% otus\_remove\_16s),]; dim (myBotu); dim (otu\_filter\_16s)

tax\_filter\_16s <- myBtaxa[rownames(otu\_filter\_16s),]

design\_filter\_16s <- droplevels(mydesign[rownames(mydesign) %in% colnames(otu\_filter\_16s),])

design\_filter\_16s <- design\_filter\_16s[colnames(otu\_filter\_16s),]

dim(otu\_filter\_16s); dim(tax\_filter\_16s); dim(design\_filter\_16s)

###### 16S sequence and OTU counts ######

sum(colSums(otu\_filter\_16s))

sort(colSums(otu\_filter\_16s))

median(colSums(otu\_filter\_16s)); mean(colSums(otu\_filter\_16s))

nrow(tax\_filter\_16s)

table(tax\_filter\_16s$Kingdom)

## Order taxonmy file by OTU

otu\_order\_16s <- match(rownames(otu\_filter\_16s), rownames(tax\_filter\_16s))

tax\_filter\_16s <- tax\_filter\_16s[otu\_order\_16s,]

otu\_filter\_16s <- otu\_filter\_16s [rowSums(otu\_filter\_16s) > 1,]

dim(otu\_filter\_16s)

sum(otu\_filter\_16s); sort(colSums(otu\_filter\_16s))

tax\_filter\_16s <- tax\_filter\_16s[rownames(otu\_filter\_16s),] ; dim(tax\_filter\_16s)

write.table(otu\_filter\_16s, file="otu\_filter\_16s.M.txt", sep="\t",quote=F)

write.table(design\_filter\_16s, file="design\_filter\_16s.M.txt", sep="\t",quote=F)

write.table(tax\_filter\_16s, file="tax\_filter\_16s.M.txt", sep="\t",quote=F)

#####**rarefy###**

otu\_rarefy\_16s <-t(otu\_filter\_16s)

(raremax\_16s <- min(rowSums(otu\_rarefy\_16s)))

set.seed(315); otu\_rarefy\_16s <- **rrarefy**(otu\_rarefy\_16s, raremax\_16s)

otu\_rarefy\_16s <- otu\_rarefy\_16s[ ,colSums(otu\_rarefy\_16s)>0] ; dim(otu\_rarefy\_16s)

otu\_rarefy\_16s<-t(otu\_rarefy\_16s)

tax\_rarefy\_16s <- tax\_filter\_16s[rownames(otu\_rarefy\_16s),] ; dim(tax\_rarefy\_16s)

design\_rarefy\_16s <- design\_filter\_16s

write.table(otu\_rarefy\_16s, file="otu\_rarefy\_16s.M.txt", sep="\t",quote=F)

write.table(design\_rarefy\_16s, file="design\_rarefy\_16s.M.txt", sep="\t",quote=F)

write.table(tax\_rarefy\_16s, file="tax\_rarefy\_16s.M.txt", sep="\t",quote=F)

##### mean relative abundance of each OTU for each sample ####

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

rep\_time <- length(colnames(otu\_16s\_RA))/ Sam.Rep ; t=1; otu\_16s\_RA\_AVE=c()

for (i in 1:rep\_time) { sub\_table= otu\_16s\_RA [,t:(t+ Sam.Rep -1)]

sub\_mean=apply(sub\_table, 1, mean)

otu\_16s\_RA\_AVE=cbind(otu\_16s\_RA\_AVE, sub\_mean)

t=t+ Sam.Rep }

colnames(otu\_16s\_RA\_AVE) <- unique (as.factor(design\_rarefy\_16s$Trt))

**write.table**(otu\_16s\_RA\_AVE, file="otu\_16s\_RA\_AVE.M.txt", sep="\t",quote=F)

########identifying FOL-altered OTUs in cultivar D72#############

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

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

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

otu\_filter\_16s<-read.table("otu\_filter\_16s.M.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

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

design.FC <- **filter**(design, FOL %in% c("Control", "Sys"));

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')) # 默认 pAdjustMethod="BH"

res

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

**# write.table**(res1, 'control\_treat.DESeq2.txt', col.names = NA, sep = '\t', quote = 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) )

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

# S.Taxa<-tax\_filter\_16s[rownames(T\_enrich), ]; rownames(S.Taxa)[ S.Taxa $Genus=="Bacillus" ]

## 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

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

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

MA.D72.Sys.FOL <- 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.FOL

########identifying FOL-altered OTUs in cultivar Z19#############

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

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

otu\_filter\_16s<-read.table("otu\_filter\_16s.M.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

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

design.FC <- **filter**(design, FOL %in% c("Control", "Sys"))

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')) # 默认 pAdjustMethod="BH"

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.FOL.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")+

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

########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.M.txt", row.names=1, sep="\t", header=T, blank.lines.skip=F, check.names=F)

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

otu\_filter\_16s<-read.table("otu\_filter\_16s.M.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

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

design.FC <- **filter**(design, FA %in% c("Control", "Sys"));

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')) # 默认 pAdjustMethod="BH"

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.FA.txt', col.names = NA, sep = '\t', quote = FALSE)

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

MA.D72.Sys.FA <- 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(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.FA

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

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

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

otu\_filter\_16s<-read.table("otu\_filter\_16s.M.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

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

design.FC <- **filter**(design, FA %in% c("Control", "Sys"));

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')) # 默认 pAdjustMethod="BH"

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.FA.txt', col.names = NA, sep = '\t', quote = FALSE)

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

MA.Z19.Sys.FA <- 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(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.FA

#############Plotting the bubble plot####

library(dplyr); library(pheatmap); library(RColorBrewer) ; library(reshape2)

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

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

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

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

Pct <- 0.01; BM=50

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

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

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

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

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

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

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

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

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

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

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

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

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

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

length(rownames(D72.Sys\_En.FOL))

length(rownames(D72.Sys\_En.FA))

D72.En = intersect( rownames(D72.Sys\_En.FOL), rownames(D72.Sys\_En.FA) )

length (D72.En)

length(rownames(Z19.Sys\_En.FOL))

length(rownames(Z19.Sys\_En.FA))

Z19.En = intersect( rownames(Z19.Sys\_En.FOL), rownames(Z19.Sys\_En.FA) )

length (Z19.En)

length(rownames(D72.Sys\_De.FOL))

length(rownames(D72.Sys\_De.FA))

D72.De=intersect( rownames(D72.Sys\_De.FOL), rownames(D72.Sys\_De.FA) )

length (D72.De)

length(rownames(Z19.Sys\_De.FOL))

length(rownames(Z19.Sys\_De.FA))

Z19.De=intersect( rownames(Z19.Sys\_De.FOL), rownames(Z19.Sys\_De.FA) )

length (Z19.De)

Cs.D72 = union(D72.En, D72.De)

Cs.Z19 = union(Z19.En, Z19.De)

csO = union(Cs.Z19, Cs.D72)

intersect(rownames(Z19.Sys\_De.FOL), csO)

intersect(rownames(Z19.Sys\_De.FA), csO)

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

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

otu\_rarefy\_16s<- read.table("otu\_rarefy\_16s.M.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)

csOTUs.tax <- tax\_rarefy\_16s [csO, ]

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(t(otu\_RA))

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

Sam.Rep=4

rep\_time <- length(colnames(otu\_RA))/ Sam.Rep ; t=1; otu\_RA\_AVE=c()

for (i in 1:rep\_time) { sub\_table= otu\_RA [,t:(t+ Sam.Rep -1)]

sub\_mean=apply(sub\_table, 1, mean)

otu\_RA\_AVE=cbind(otu\_RA\_AVE, sub\_mean)

t=t+ Sam.Rep }

colnames(otu\_RA\_AVE) <- unique (as.factor(design\_rarefy\_16s$Trt))

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

otu\_RA <- as.data.frame(otu\_RA\_AVE)

otu\_RA$Phylum <- tax\_rarefy\_16s[csOTUs, ]$Phylum

otu\_RA$Class <- tax\_rarefy\_16s[csOTUs, ]$Class

otu\_RA$Order <- tax\_rarefy\_16s[csOTUs, ]$Order

otu\_RA$Family <- tax\_rarefy\_16s[csOTUs, ]$Family

otu\_RA$Genus <- tax\_rarefy\_16s[csOTUs, ]$Genus

otu\_RA$id <- rownames(otu\_RA)

otu\_RA$id <- paste(otu\_RA $Genus, otu\_RA$id, sep=";")

otu\_RA$id <- paste(otu\_RA $Family, otu\_RA $id, sep="; g:")

otu\_RA$id <- paste(otu\_RA $Order, otu\_RA $id, sep="; f:")

otu\_RA$id <- paste(otu\_RA $Class, otu\_RA $id, sep="; o:")

otu\_RA$id <- paste(otu\_RA $Phylum, otu\_RA $id, sep="; c:")

for (i in rownames(otu\_RA)) {

if (i %in% Cs.D72 ==TRUE)

{ otu\_RA[rownames(otu\_RA)==paste(i),]$id <- paste("C", otu\_RA[rownames(otu\_RA)==paste(i),]$id, sep=",") }

else

{ otu\_RA [rownames(otu\_RA)==paste(i),]$id <- otu\_RA [rownames(otu\_RA)==paste(i),]$id } }

for (i in rownames(otu\_RA)) {

if (i %in% Cs.Z19 ==TRUE)

{ otu\_RA[rownames(otu\_RA)==paste(i),]$id <- paste("B", otu\_RA[rownames(otu\_RA)==paste(i),]$id, sep=",") }

else

{ otu\_RA [rownames(otu\_RA)==paste(i),]$id <- otu\_RA [rownames(otu\_RA)==paste(i),]$id } }

otu\_RA <- otu\_RA [order(otu\_RA [, c("id")]), ] #排序, decreasing = T

rownames(otu\_RA) <- otu\_RA$id

otu\_RA <- otu\_RA [ , !colnames(otu\_RA) %in% c("Phylum", "Class", "Order", "Family", "Genus", "id")]

otu\_RA <- as.data.frame(t(otu\_RA))

csO <- colnames(otu\_RA)

otu\_RA $Trt <- unique (as.factor(design\_rarefy\_16s$Trt))

otu\_RA $Name <- rownames(otu\_RA)

otu\_RA.long = melt(otu\_RA, id.vars=c("Trt", "Name"),

measure.vars=c(csO),

variable.name='csO', value.name='Abundance')

range(otu\_RA.long$ Abundance)

#otu\_RA.long$ csOTU <- gsub("b", "", otu\_RA.long$ csOTU)

library (ggplot2)

p.RA.dot<-ggplot(otu\_RA.long, aes(x=factor(Trt, levels=c('D72-FOL-C', 'D72-FOL-S', 'D72-FA-C', 'D72-FA-S', 'Z19-FOL-C', 'Z19-FOL-S', 'Z19-FA-C', 'Z19-FA-S')), y=csO ))+

geom\_point(aes(size= Abundance), colour = "grey15")+

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

theme\_bw()+

theme(axis.text=element\_text(size=6, color="black"), axis.title= element\_blank(),

legend.text=element\_text(size=8), legend.background=element\_rect(fill='NA'),

# axis.text.x = element\_text(vjust = 0.5, hjust = 0.5, angle = 90), #axis.text.y = element\_text(size=8),

# axis.text.y=element\_blank(), axis.ticks.y= element\_blank (),

panel.background = element\_blank(), panel.grid.major=element\_line(colour=NA),

panel.grid.minor=element\_line(colour=NA) )

p.RA.dot

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

Cs.D72 = union(D72.En, D72.De)

Cs.D72.data <- forMA.D72.FOL[Cs.D72, c("log2FoldChange", "padj") ]

colnames(Cs.D72.data) <- c("FC.D72.FOL", "P.72")

Cs.D72.data$ FC.D72.FA <- forMA.D72.FA [Cs.D72, ]$ log2FoldChange

Cs.D72.data <- Cs.D72.data [ , -which(colnames(Cs.D72.data) %in% c("P.72"))]

range(Cs.D72.data)

bk <- c(seq(-2.3, -0.1, by=0.01), seq(0, 2.3,by=0.01))

pheatmap(Cs.D72.data, clustering\_distance\_rows ="euclidean", # "euclidean", "maximum", "manhattan", "canberra", "binary" "minkowski"

clustering\_distance\_cols ="euclidean",

clustering\_method ="complete",

color=c(colorRampPalette(colors=c("dodgerblue2","white"))(length(bk)/2),colorRampPalette(colors=c("white","red3"))(length(bk)/2)),

legend\_breaks=seq(-2, 2, 1), breaks=bk,

fontsize=6, fontsize\_row=8)

Cs.Z19 = union(Z19.En, Z19.De)

Cs.Z19.data <- forMA.Z19.FOL[Cs.Z19, c("log2FoldChange", "padj") ]

colnames(Cs.Z19.data) <- c("FC.Z19.FOL", "P.19")

Cs.Z19.data$ FC.Z19.FA <- forMA.Z19.FA [Cs.Z19, ]$ log2FoldChange

Cs.Z19.data <- Cs.Z19.data [ , -which(colnames(Cs.Z19.data) %in% c("P.19"))]

range(Cs.Z19.data)

bk <- c(seq(-2.3, -0.1, by=0.01), seq(0, 2.3,by=0.01))

pheatmap(Cs.Z19.data, clustering\_distance\_rows ="euclidean", # "euclidean", "maximum", "manhattan", "canberra", "binary" "minkowski"

clustering\_distance\_cols ="euclidean",

clustering\_method ="complete",

color=c(colorRampPalette(colors=c("dodgerblue2","white"))(length(bk)/2),colorRampPalette(colors=c("white","red3"))(length(bk)/2)),

legend\_breaks=seq(-2, 2, 1), breaks=bk,

fontsize=6, fontsize\_row=8)