library (vegan); library (dplyr); library(phyloseq); library(ape); library(ggpubr); library(ggplot2); library (ggrepel); library(edgeR)

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

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

identical(rownames(myBasv), rownames(myBtaxa))

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

identical(colnames(myBasv), rownames(mydesign))

Sam.Rep = 4

myBtaxa$labels <- myBtaxa$Phylum

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

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

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

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

asv\_filter\_16s <- myBasv[-which(rownames(myBasv) %in% asvs\_remove\_16s),]; dim (myBasv); dim (asv\_filter\_16s)

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

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

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

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

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

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

asv\_filter\_16s <- asv\_filter\_16s [rowSums(asv\_filter\_16s) > 20,]

dim(asv\_filter\_16s)

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

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

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

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

set.seed(123)

asv\_1 = otu\_table(asv\_filter\_16s, taxa\_are\_rows = T)

asv\_2 = phyloseq(asv\_1)

asv\_Flattening1 = rarefy\_even\_depth(asv\_2,replace = TRUE)

asv\_rarefy\_16s = as.data.frame(asv\_Flattening1@.Data )

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

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

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

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

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

####ploting Rarefaction curves ####

par (mar=c(5,4,4,2))

asv\_filter\_16s <- as.matrix(asv\_filter\_16s)

raremax\_16s <- min(colSums(asv\_filter\_16s))

rarecurve (t(asv\_filter\_16s), step=250, col="black", cex=0.6, label=F, ylim=c(0, 1600),

xlab="Number of reads sampled", ylab="Number of ASVs") ##

abline(v=raremax\_16s, lty=2, col="dodgerblue", lwd=2)

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

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

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

dim(asv\_rarefy\_16s)

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)

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

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

dim(asv\_filter\_16s)

##### TMM normalize 16S counts for whole community beta diversity analysis #####

group\_16s <- design\_filter\_16s$Trt

edgeR\_16s<- DGEList(counts= asv\_filter\_16s,

group=design\_filter\_16s$Trt,

genes=tax\_filter\_16s)

edgeR\_16s <- calcNormFactors(edgeR\_16s)

asv\_norm\_16s <- **cpm**(edgeR\_16s, normalized.lib.sizes=T,log=F)

**write.table**(asv\_norm\_16s, file="asv\_norm\_16s.txt", sep="\t",quote=F)

design <- design\_rarefy\_16s; asv.data <- asv\_norm\_16s

distance.w <- vegdist(**decostand**(t(asv.data), "hellinger"), method = 'bray')

res.w <- **pcoa**(distance.w)

res.v.w <- res.w$vectors

site.w <- data.frame(x= res.v.w[,1], y= res.v.w[,2], Trt=design$Trt )

site.w$Trt <- as.factor(site.w$Trt)

site.w$sample <- rownames(site.w)

write.table(site.w, file="PCoACoord.txt", sep="\t",quote=F)

B.PCoA.Bray <- ggplot(data=site.w, aes(x, y, colour = Trt))+geom\_point(size=4) + #, shape=Trt

xlab(paste("PCoA1", paste("(",round(res.w$values[1,2]\*100,2),"%",")",sep=""),sep=" ")) +

ylab(paste("PCoA2", paste("(",round(res.w$values[2,2]\*100,2),"%",")",sep=""),sep=" ")) +

**stat\_ellipse**(aes(group=Trt, fill=Trt), type="norm", linewidth=0, alpha=0.2, level=0.80, geom="polygon", lty="dashed", color=NA) +

ylim(-0.18, 0.25) +

xlab(paste("PCoA1", paste("(",round(res.w$values[1,2]\*100,2),"%",")",sep=""),sep=" "))+

ylab(paste("PCoA2", paste("(",round(res.w$values[2,2]\*100,2),"%",")",sep=""),sep=" ")) +

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() )

B.PCoA.Bray

design$Trt <- as.factor (design$Trt); adonis2(distance.w ~ Trt, data=design, permutations=9999)

design.1 <- **filter(design, Trt == "W-" | Trt == "W+")**;

design.1 <- **filter**(design.1, Trt !="Restored"); asv.data.1 <- asv.data[, rownames(design.1)]

distance.w.1 <- vegdist(**decostand**(t(asv.data.1), "hellinger"), method = 'bray')

design.1$Trt <- as.factor (design.1$Trt); adonis2(distance.w.1 ~ Trt, data=design.1, permutations=9999)

design.1 <- **filter(design, Trt == "Jai1-" | Trt == "Jai1+")**;

design.1 <- **filter**(design.1, Trt !="Restored"); asv.data.1 <- asv.data[, rownames(design.1)]

distance.w.1 <- vegdist(**decostand**(t(asv.data.1), "hellinger"), method = 'bray')

design.1$Trt <- as.factor (design.1$Trt); adonis2(distance.w.1 ~ Trt, data=design.1, permutations=9999)

design.1 <- **filter(design, Trt == "Spr8-" | Trt == "Spr8+")**;

design.1 <- **filter**(design.1, Trt !="Restored"); asv.data.1 <- asv.data[, rownames(design.1)]

distance.w.1 <- vegdist(**decostand**(t(asv.data.1), "hellinger"), method = 'bray')

design.1$Trt <- as.factor (design.1$Trt); adonis2(distance.w.1 ~ Trt, data=design.1, permutations=9999)

##### relative abundances for specific genus####

library(ggplot2); library(ggpubr); library(reshape2); library (dplyr); library(car); library(lsmeans); library(multcomp)

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

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

dim(myBASV); dim(myBtaxa)

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

ASV\_its\_RA <- t(t(myBASV)/colSums(myBASV))\*100

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

length(PHYLAnames\_its)

y <- NULL

ASVnames <- rownames(ASV\_its\_RA)

for (i in PHYLAnames\_its){

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

y <- rbind(y,x) }

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

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

CLASS\_mat\_its <- y

colSums(CLASS\_mat\_its)

dim(CLASS\_mat\_its)

S.Genus<- as.data.frame( t(CLASS\_mat\_its [ c("Streptomyces", "Actinoplanes" ), ]) )

S.Genus$Trt <- design $ Trt

**write.table**(S.Genus, file="S.Genus.JAMu.txt", sep="\t",quote=F)

p.Streptomyces <- **ggplot**(S.Genus, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= Streptomyces, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 9), expand=c(0, 0.0001)) +

labs(x=NULL, y="Streptomyces sp. (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

p.Streptomyces

p.Actinoplanes <- **ggplot**(S.Genus, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= Actinoplanes, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 3.2), expand=c(0, 0.0001)) +

labs(x=NULL, y="Actinoplanes sp. (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

p.Actinoplanes

**ggarrange**(p.Streptomyces, p.Actinoplanes, nrow=1, ncol=2, align="hv", legend="none" ) #"right" common.legend = T,

##### relative abundances for specific ASVs####

library(ggplot2); library(ggpubr); library(reshape2); library (dplyr); library(car); library(lsmeans); library(multcomp)

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)

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

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

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

dim(ASV\_filter\_16s)

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

design\_filter\_16s $ ASV\_71 <- ASV\_16s\_RA[c("ASV\_71"), ]

design\_filter\_16s $ ASV\_8 <- ASV\_16s\_RA[c("ASV\_8"), ]

design\_filter\_16s $ASV\_10 <- ASV\_16s\_RA[c("ASV\_10"), ]

design\_filter\_16s $ ASV\_36 <- ASV\_16s\_RA[c("ASV\_36"), ]

mydata <- design\_filter\_16s

write.table(mydata, file="S.ASV.txt", sep="\t",quote=F)

P.ASV\_71 <- **ggplot**(mydata, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= ASV\_71, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 0.61), expand=c(0, 0.0001)) +

labs(x=NULL, y="ASV71 (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

P.ASV\_71

P.ASV\_8 <- **ggplot**(mydata, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= ASV\_8, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 2.3), expand=c(0, 0.0001)) +

labs(x=NULL, y="ASV8 (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

P.ASV\_8

P.ASV\_10 <- **ggplot**(mydata, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= ASV\_10, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 2.3), expand=c(0, 0.0001)) +

labs(x=NULL, y="ASV10 (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

P.ASV\_10

P.ASV\_36 <- **ggplot**(mydata, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+')) , y= ASV\_36, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35, coef = 5) +

**geom\_boxplot**(alpha=0, width=0.5, lwd=1, fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2.5, alpha=0.70) +

scale\_y\_continuous (limits=c(0, 1.1), expand=c(0, 0.0001)) +

labs(x=NULL, y="ASV36 (RA %)") +

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

P.ASV\_36

**ggarrange**(P.ASV\_71, P.ASV\_8, P.ASV\_10, P.ASV\_36, nrow=1, ncol=4, align="hv", legend="none" )

model<-lm(ASV\_71 ~Trt, data= mydata); #summary(model); #Anova(model, type="II")

marginal=lsmeans(model, ~Trt); #pairs(marginal, adjust="fdr")

cld(marginal, alpha=0.05, Letters=letters, adjust="fdr", reversed=T)

######### rhizosphere transplant for the mutant experiment####

library(ggplot2); library(ggpubr); library (dplyr); library(car); library(lsmeans); library(multcomp)

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

JMTrans.Lesion.box <- **ggplot**(mydata, aes(x= **factor**(Trt, levels=c('W-', 'W+', 'Jai1-', 'Jai1+', 'Spr8-', 'Spr8+' )), y=Area, color=Trt) ) +

**stat\_boxplot** (geom="errorbar", width=0.35) +

**geom\_boxplot**(alpha=0, outlier.size=0.75, outlier.shape=1, width=0.5, lwd=1,

outlier.color = "grey30", fill="transparent", show.legend = F) +

**geom\_jitter(**color="grey30", position=position\_jitter(0.22), size=2, alpha=0.70) +

labs(x=NULL, y="Lesion area (mm2)")+

scale\_y\_continuous (limits=c(0, 155), expand=c(0, 0.0001))+

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=12, color="black"),

axis.title=element\_text(size=12),

axis.text.x = element\_text(angle = 75, hjust = 0.5, vjust = 0.5, color="black"),

legend.position = "none" )

JMTrans.Lesion.box

model<-lm(Area ~Trt, data= mydata); #summary(model); #Anova(model, type="II")

marginal=lsmeans(model, ~ Trt); #pairs(marginal, adjust="fdr")

cld(marginal, alpha=0.05, Letters=letters, adjust="fdr", reversed=T)