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

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

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

identical(rownames(myBasv), rownames(myBtaxa)) #TRUE

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

Sam.Rep=4

myBtaxa$labels <- myBtaxa$Phylum

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

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

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

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

library(phyloseq)

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)); raremax\_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\_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)

#############Plotting####################

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) +

xlim(-0.21, 0.22) +

ylim(-0.35, 0.35) +

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)

####

library(ggplot2); library(ggpubr); library(reshape2); library (dplyr) ; library(ggbeeswarm)

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)

#####Genus relative abundance#####

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

head(CLASS\_mat\_its)

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.Mixture.txt", sep="\t",quote=F)

pData.long = melt(S.Genus, id.vars=c("Trt"),

measure.vars= c("Streptomyces", "Actinoplanes"),

variable.name='Genus', value.name='RA')

PS.Genus <- ggplot(pData.long, aes(x = Genus, y= RA , color = Trt )) +

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

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

**geom\_beeswarm**(dodge.width = 0.36, alpha=0.70, size=2 ) +

scale\_y\_continuous (limits=c(0, 3.5), expand=c(0, 0.000001))+

labs(x=NULL, y="Relative abundance (%)")+

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=9, color="black"),

legend.position="none", axis.title=element\_text(size=12) )

PS.Genus

**stats::t.test**(S.Genus $ Streptomyces ~ S.Genus $Trt, p.adj="BH", var.equal=F)

**stats::t.test**(S.Genus $ Actinoplanes ~ S.Genus $Trt, p.adj="BH", var.equal=F)

#######

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

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 $ ASV7 <- asv\_16s\_RA[c("ASV7"), ]

design\_filter\_16s $ ASV74 <- asv\_16s\_RA[c("ASV74"), ]

design\_filter\_16s $ ASV191 <- asv\_16s\_RA[c("ASV191"), ]

design\_filter\_16s $ ASV43 <- asv\_16s\_RA[c("ASV43"), ]

design\_filter\_16s $ ASV235 <- asv\_16s\_RA[c("ASV235"), ]

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

pData.long = melt(design\_filter\_16s, id.vars=c("Trt"),

measure.vars= c("ASV7", "ASV74", "ASV191", "ASV43", "ASV235"),

variable.name='ASV', value.name='RA')

ASV.Com <- ggplot(pData.long, aes(x = ASV, y= RA , color = Trt )) +

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

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

**geom\_beeswarm**(dodge.width = 0.36, alpha=0.70, size=2.5 ) +

scale\_y\_continuous (limits=c(0, 1.2), expand=c(0, 0.000001))+

labs(x=NULL, y="Relative abundance (%)")+

theme\_bw() +

theme(panel.grid=element\_blank(), axis.text=element\_text(size=9, color="black"),

legend.position="none", axis.title=element\_text(size=12) )

ASV.Com

mydata <- design\_filter\_16s

**stats::t.test**(mydata $ ASV7 ~ mydata $Trt, p.adj="BH", var.equal=F)

**stats::t.test**(mydata $ ASV74 ~ mydata $Trt, p.adj="BH", var.equal=F)

**stats::t.test**(mydata $ ASV191 ~ mydata $Trt, p.adj="BH", var.equal=F)

**stats::t.test**(mydata $ ASV43 ~ mydata $Trt, p.adj="BH", var.equal=F)

**stats::t.test**(mydata $ ASV235 ~ mydata $Trt, p.adj="BH", var.equal=F)