library(vegan); library(ape); library(ggpubr); library(ggplot2); library (readr); library(phyloseq); library(GUniFrac); library(stringr); library (ggrepel); library (dplyr)

design\_rarefy\_16s <- read.table("design\_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)

dim(otu\_rarefy\_16s)

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

library (readr); library(edgeR); library(phyloseq)

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)

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

## Apply TMM normalization to entire 16S data set and create phyloseq objects for later analysis

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

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

group=design\_filter\_16s$Trt,

genes=tax\_filter\_16s)

edgeR\_16s <- calcNormFactors(edgeR\_16s)

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

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

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

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

distance.w <- vegdist(**decostand**(t(otu.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)

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

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

geom\_text\_repel(data=site.w, aes(x, y, label=sample), colour="black", size=5) +

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=NA, alpha=0.2, level=0.75, geom="polygon", lty="dashed", color="black") +

scale\_color\_manual( values= c("dodgerblue", "darkorange"), labels=c("CK", "VOC") ) + #, guide="none"

scale\_fill\_manual(values= c("dodgerblue", "darkorange"), guide="none") +

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)