16S rRNA seq processing SHIME

Contents

1 Initial settings

```
# set the version of the SILVA release that was used for classification
silvadb <- 138
#set the version of the RDP release that was used
# rdprel <- 16

#relative location from the report project to the data.
fldr <-"/media/projects2/LisaM/SHIME_EM/Illumina/fastq3/"
dataname <-"SHIME"
#relative path (from fldr) to select other Metadata files
metadatafn <-"/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Metadata.xlsx"
```

2 Load packages

```
library(BiocParallel)
library(multcompView, lib.loc = "/usr/local/lib/R/site-library")
library(pander)
library(NMF)
library(knitr)#for simple markdown tables

library(devtools) #nicer session info
# library(CMETNGS) #will be integrated into CMETNGS package

library(xlsx)#handling Excel files
library(openxlsx)#handling Excel files without Java deps
library(readxl) #faster handling of excel files

library(parallel)#parallel computation in R
```

```
library(purrr) #functional programming

library(data.table) #data wrangling
library(plyr)
library(dplyr) #data wrangling (mapvalues)
suppressPackageStartupMessages(library(dplyr))#data wrangling
library(tidyr)#tidy data
library(reshape2)#tidy data
suppressPackageStartupMessages(library(reshape2))#for melt function
library(splitstackshape) #for csplit

library(vegan)#for ecological calculations
library(phyloseq)#for microbiome census data processing
library(Phenoflow)
library(SPECIES)#alpha diversity estimators
library(ade4)#for ecological calculations

library(ggplot2)
library(ggbiplot)
library(ggplotify)#for adequate plotting
library(scales)#for percent formatting
library(gtools)
library(ggsignif)
library(ggpubr)
library(ggfortify)
library(ggrepel)
library(cowplot)
library(gplots)#for heatmap.2
library(grid)
library(gridExtra)

library(RColorBrewer)
library(viridis) # for color-blind nice color palettes

library(xtable)#for more advanced tables
#library(ape)#dealing with phylogenetic trees

library(DESeq2) #normalization procedures to cope with differences in smp depth
```

3 Source Links

4 User defined functions

5 Load data

```
#Read in taxonomy
if(.Platform$OS.type == "unix") {
  crfn <- system(paste("ls",fldr," | grep contigs.report"),intern=TRUE)
} else {
```

```
  filelist <- list.files(fldr)
  crfn <- grep(".*contigs.report",filelist,value=TRUE)
}
fn <- sub(".contigs.report","",crfn)
ini <- fread(paste(fldr,"/",crfn,sep=""),header=TRUE)
csum <- fread(paste(fldr,"/",fn,
                    ".trim.contigs.summary",sep=""),header=TRUE)
firsttrimsum <- fread(paste(fldr,"/",fn,
                            ".trim.contigs.good.summary",sep=""),
                      header=TRUE)
uniquesum <- fread(paste(fldr,"/",fn,
                         ".trim.contigs.good.unique.summary",sep=""),
                   header=TRUE)
postalnsum <- fread(paste(fldr,"/",fn,
                          ".trim.contigs.good.unique.good.filter.summary",sep=""),
                    header=TRUE)
preclussum <- fread(paste(fldr,"/",fn,
              ".trim.contigs.good.unique.good.filter.unique.precluster.summary",sep=""),
                    header=TRUE)
postuchimeclasssum <- fread(paste(fldr,"/",fn,
              ".trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.summary",
              sep=""), header=TRUE)
otutaxonomy <- fread(paste(fldr,"/",fn,
".trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.opti_mcc.0.03.cons.taxonomy",
              sep=""),header=TRUE)

taxonomy.spl <- preformattax.new(otutaxonomy) #spl = split  #np = no probabilities
taxonomy.np <- taxonomy.spl %>% dplyr::select(-dplyr::contains("Prob"))

#-------------------------------------------------------------------------------------------#
# otu.table.first.attempt <- otu.table

# read in reads per sample = OTU-table
otu.table <- fread(paste(fldr,"/",fn,
".trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.opti_mcc.shared",
              sep=""),header=TRUE)
otu.table <- as.data.frame(otu.table)
otu.table <- otu.table[,-otu.table$label]

desgroups <- otu.table$Group
otu.table.x <- otu.table[,3:ncol(otu.table)]# OTU-table wihout groupnr etc
rownames(otu.table.x) <- desgroups
otu.table.t <- as.data.frame(t(otu.table.x))# transposed: OTU's are rows, samples are columns
otu.table.t.ns <- otu.table.t[which(rowSums(otu.table.t)!=1),]# ns = no singletons
otu.table.ns <- data.frame(label=rep(0.03,ncol(otu.table.t.ns)),
                       Group=colnames(otu.table.t.ns),
                       numOtus=nrow(otu.table.t.ns),t(otu.table.t.ns))

# suppressWarnings(try(write.table(x=otu.table.ns,file=paste(fldr,"/sharedns.shared",
# sep=""),row.names=FALSE,quote=FALSE),silent=TRUE))

#filter taxonomy (i.e. remove singletons from taxonomy)
taxonomy.np.ns <- taxonomy.np[which(rownames(taxonomy.np)
```

```
                                    %in% rownames(otu.table.t.ns)),]
```

6 Load Metadata

```
metadata <- readxl::read_excel("Metadata.xlsx",sheet = "Metadata")
metadata <- as.data.frame(metadata) #to avoid warnings/errors with rownames

# if(is.numeric(metadata$SampleName)) #check for fully numeric sample names
# {
#   #metadata$SampleName <- paste(metadata$EM,metadata$EM_Conc,metadata$Donor,
#   #metadata$Timepoint,sep=" - ")
#   metadata$SampleName <- as.character(metadata$SampleName)
# }
metadata$SHIME <- as.factor(metadata$SHIME)
metadata$Donor <- as.factor(metadata$Donor)
metadata$Timepoint <- as.factor(metadata$Timepoint)
metadata$Emulsifier <- as.factor(metadata$Emulsifier)
metadata$EM_Conc <- as.factor(metadata$EM_Conc)
```

7 Loading SCFA Data

```
#data.VFA <- read.csv2('SCFA_All_Donors.csv', header= TRUE)
data.SCFA <- readxl::read_xlsx("20201006_SHIME_SCFA.xlsx",sheet = "Together_mmol")
colnames(data.SCFA)[5:15] <- c("Acetic_acid","Propionic_acid","Isobutyric_acid",
                               "Butyric_acid","Isovaleric_acid","Valeric_acid",
                               "Isocaproic_acid","Caproic_acid","ISTD","Heptanoic_acid",
                               "Octanoid_acid")
data.SCFA$Donor <- data.SCFA$SHIME -1
data.SCFA <- unite(data.SCFA, Treatment, c(Emulsifier, EM_Conc), remove=FALSE, sep =' - ')
data.SCFA <- unite(data.SCFA, TreatmentTime, c(Treatment, Timepoint), remove=FALSE,
                   sep =' - ')
data.SCFA <- unite(data.SCFA, TreatmentDonor, c(Treatment, Donor), remove=FALSE,
                   sep =' - D')

data.SCFA$Emulsifier<- as.factor(data.SCFA$Emulsifier)
data.SCFA$Treatment<- as.factor(data.SCFA$Treatment)
data.SCFA$TreatmentTime<- as.factor(data.SCFA$TreatmentTime)
data.SCFA$EM_Conc<- as.factor(data.SCFA$EM_Conc)
data.SCFA$Donor<- as.factor(data.SCFA$Donor)
```

8 Rarefaction curves raw otu-data

9 Filtering

```
# here, positive controles, RS-damples and B-samples are filtered out of the OTU-table,
# before further assessment
```

```
otu.table.t.ns <- otu.table.t.ns[, -grep("B", colnames(otu.table.t.ns))]

#also filtering metadata
metadata <- metadata[-grep("B",metadata$Samplenr),]
metadata <- unite(metadata, Treatment, c(Emulsifier, EM_Conc), remove=FALSE, sep =' - ')
metadata <- unite(metadata, TreatmentTime, c(Treatment, TimeCode), remove=FALSE, sep =' - ')
```

10 calculations

```
#use taxonomy file to destinguish how many phyla, families and genera were detected.

n_phyla <- length(unique(taxonomy.np.ns$Phylum))
n_classes <- length(unique(taxonomy.np.ns$Classis))
n_order <- length(unique(taxonomy.np.ns$Ordo))
n_family <- length(unique(taxonomy.np.ns$Familia))
n_genus <- length(unique(taxonomy.np.ns$Genus))
```

11 Plotting relative abundances

```
#reordering columns in increasing order of samplenames.
otu.table.t.ns <- otu.table.t.ns[, as.character(metadata$Samplenr)]
#sort otu.table.t.ns following samplenummer
otu.table.t.ns.cpnr <- otu.table.t.ns

otu.table.t.ns.cpnr <- otu.table.t.ns.cpnr[,as.character(metadata$Samplenr)]

# calculation of relative abundances (instead of absolute reads)
otu.table.t.ns.cpnr.relabun <- decostand(data.frame(otu.table.t.ns.cpnr[1:142]), 2,
                                        method = "total")
colnames(otu.table.t.ns.cpnr.relabun) <- colnames(otu.table.t.ns.cpnr[1:142])

#controle
otu.sums <- colSums(otu.table.t.ns.cpnr.relabun)
```

11.1 Genus

```
# add taxonomy column
otu.table.t.ns.cpnr.relabun$Genus <- taxonomy.np.ns$Genus

#summate relative abundances in each sample per genus
otu.table.t.ns.cpnr.relabun.Gr <- aggregate(otu.table.t.ns.cpnr.relabun[1:142],
                                    by=list(Category=otu.table.t.ns.cpnr.relabun$Genus),
                                    FUN=sum)

#filter out 10 most abundant genusses
#take rowsums
otu.table.t.ns.cpnr.relabun.Gr$Sums <- rowSums(otu.table.t.ns.cpnr.relabun.Gr[2:143])
```

```
#order columns on value of the rowsums
otu.table.t.ns.cpnr.relabun.Gr <- otu.table.t.ns.cpnr.relabun.Gr[order(
  -otu.table.t.ns.cpnr.relabun.Gr$Sums),]
#take first 8 Genera's
otu.table.t.ns.cpnr.relabun.Gr.top8 <- otu.table.t.ns.cpnr.relabun.Gr[1:10,]
#add 11th row as a sum of all the rest
otu.table.t.ns.cpnr.relabun.Gr.top8[11,2:144] <- colSums(
  otu.table.t.ns.cpnr.relabun.Gr[11:199,2:144])

#adding "Other" on nineth place of first column
otu.table.t.ns.cpnr.relabun.Gr.top8$Category <- as.character(
  otu.table.t.ns.cpnr.relabun.Gr.top8$Category)
otu.table.t.ns.cpnr.relabun.Gr.top8$Category[11] <- c("Other")
otu.table.t.ns.cpnr.relabun.Gr.top8$Category <- as.factor(
  otu.table.t.ns.cpnr.relabun.Gr.top8$Category)

otu.table.t.ns.cpnr.relabun.Gr.top8$Sums <- NULL  #remove sums column

#rearrange data-table in order to be able to add metadata
otu.table.t.ns.cpnr.relabun.Gr.top8.R <- tidyr::gather(
  otu.table.t.ns.cpnr.relabun.Gr.top8,key = "Samplenr", value = "RelativeAbundance",2:143)
otu.table.t.ns.cpnr.relabun.Gr.top8.R$Samplenr <- as.factor(
  otu.table.t.ns.cpnr.relabun.Gr.top8.R$Samplenr)

#add metadata
otu.merge.genus <- merge(otu.table.t.ns.cpnr.relabun.Gr.top8.R,metadata, by = "Samplenr")
#otu.merge.genus <- dplyr::arrange(otu.merge.genus,as.numeric(otu.merge.genus$Samplenr))
#arrange will not work to order in ascending order, since not all samplenumbers are numbers
otu.merge.genus$Samplenr <- gdata::reorder.factor(otu.merge.genus$Samplenr,
                                                  new.order=metadata$Samplenr)
#This works to arrange the dataframe in ascending order of numbers

write.xlsx(otu.merge.genus,"OTU10_Relative_Abundances.xlsx", append = TRUE,
           sheetName = "Otu_merge_genus")
```

11.1.1 Explorative barplots

```
#we're just plotting all samples as stacked bars next to eachother in one plot
X.order <- metadata$Samplenr[1:142]

Exploreplot <- ggplot(otu.merge.genus, aes(x = Samplenr,y = RelativeAbundance, fill = factor(Category,l
  geom_bar(stat="identity",position = position_fill(reverse = TRUE)) +
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999")) +
  # scale_fill_brewer(palette="Set1")+
  scale_x_discrete(limits=X.order)+ theme_minimal() + labs(fill = "Top 10 Genera")+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
Exploreplot
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_ExplorativeBarplot1.t
```

```
     width=16*dpi,height=8*dpi,res=dpi,compression="lzw")
Exploreplot
dev.off()

#lets tratify for lumen vs mucin
# otu.merge.genus.nP.nR <- otu.merge.genus[-grep("N",otu.merge.genus$Emulsifier),]
# otu.merge.genus.nP.nR$TreatmentTime <- as.factor(otu.merge.genus.nP.nR$TreatmentTime)

# X.order.2 <- c("Control - 0","P80 - 0.05","P80 - 0.5", "SoyLec - 0.05","SoyLec - 0.5",
#                "RL - 0.05","RL - 0.5")

Exploreplot2 <- ggplot(otu.merge.genus, aes(x = TreatmentTime,y = RelativeAbundance,
                                            fill = factor(Category,levels=as.character(otu.table.
  geom_bar(stat="identity",position = position_fill(reverse = TRUE)) +
  facet_grid(SHIME~Location, drop =TRUE)+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999")) +
  # scale_fill_brewer(palette="Set1")+
  # scale_x_discrete(limits=X.order.2)+
  theme_minimal() + labs(fill = "Top 10 Genera")+ theme(axis.text.x=element_text(
    angle=90, hjust=1, vjust = 0,size = 8))
Exploreplot2
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_ExplorativeBarplot1.t
     width=16*dpi,height=8*dpi,res=dpi,compression="lzw")
Exploreplot2
dev.off()
```

11.1.2 Plot Lumen

```
otu.merge.genus.lumen <- dplyr::filter(otu.merge.genus, Location == 'lumen')
otu.merge.genus.lumen$Treatment <- as.factor(otu.merge.genus.lumen$Treatment)

#plot bargraphs
# BarOrder <- c("Other","Ruminococcaceae_unclassified","Roseburia","Prevotella",
#"Lachnospiraceae_unclassified","Faecalibacterium","Escherichia/Shigella","Bacteroides",
#"Alistipes")


## GENERAL PLOT
barplot.genus.lumen <-ggplot(data=otu.merge.genus.lumen, aes(x= Samplenr,
                                                             y=RelativeAbundance,
                                                             fill = factor(Category, levels=as.characte
  geom_bar(stat="identity",position = position_fill(reverse = TRUE))+
  labs(fill = "Top 8 Genera")+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+
  theme_minimal()+ theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.genus.lumen
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumen1.ti
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.genus.lumen
dev.off()


X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".", "P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.2 <-ggplot(data=otu.merge.genus.lumen, aes(x= TreatmentTime,
                                                               y=RelativeAbundance,
                                                               fill = factor(Category, levels=as.charac
  geom_bar(stat="identity", position = position_fill(reverse = TRUE))+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
  # axis.text.y=element_text(size = 6))+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  # legend.text=element_text(size=7))
barplot.genus.lumen.2
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumen2.ti
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.2
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
    barplot.genus.lumen.2
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.lumen.2
dev.off()


#P80 & RHANOLIPIDS ------------------------------------------------------------------------#

otu.merge.genus.lumen.RL.P80 <- dplyr::filter(otu.merge.genus.lumen, Emulsifier %in%
                                                c("Control","P80","RL","Fecal"))
otu.merge.genus.lumen.RL.P80$Treatment <- as.factor(otu.merge.genus.lumen.RL.P80$Treatment)

#plot bargraphs
# BarOrder <- c("Other","Ruminococcaceae_unclassified","Roseburia","Prevotella",
#"Lachnospiraceae_unclassified","Faecalibacterium","Escherichia/Shigella","Bacteroides",
#"Alistipes")

X.order <- c("Fecal - 0","Control - 0","P80 - 0.05","P80 - 0.5", "SoyLec - 0.05",
             "SoyLec - 0.5","RL - 0.05","RL - 0.5")

## GENERAL PLOT
barplot.genus.lumen.RL.P80 <-ggplot(data=otu.merge.genus.lumen.RL.P80,aes(x= Samplenr,
                                                                          y=RelativeAbundance,
                                                                          fill = factor(Category, levels=as.
  geom_bar(stat="identity",position = position_fill(reverse = TRUE)) +
  labs(fill = "Top 8 Genera")+scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57",
                                                           "#A660CC", "#E1A13F", "#F781BF",
                                                           "#E8D241", "#A65628","#400F73",
                                                           "#FA815F","#999999"))+
  theme_minimal()+ theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.genus.lumen.RL.P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumenRLP8
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.RL.P80
dev.off()

X.order.lumen.RL.P80 <- c("Fecal - 0 - 0",".",
             "Control - 0 - 0","Control - 0 - 1","Control - 0 - 3","Control - 0 - 7",".",
             "P80 - 0.05 - 0","P80 - 0.05 - 1","P80 - 0.05 - 3","P80 - 0.05 - 7",".",
             "P80 - 0.5 - 0","P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")
```

```
barplot.genus.lumen.RL.P80 <-ggplot(data=otu.merge.genus.lumen.RL.P80,aes(x=TreatmentTime,
                                                                          y=RelativeAbundance,
                                                                        fill = factor(Category, levels=as.cha
  geom_bar(stat="identity", position = position_fill(reverse = TRUE)) +
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.RL.P80)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.lumen.RL.P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumenRLP8
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.RL.P80
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
   barplot.genus.lumen.RL.P80
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.lumen.RL.P80
dev.off()
```

11.1.3 Plot Mucus

```
otu.merge.genus.mucus <- dplyr::filter(otu.merge.genus, Location == 'mucin')
otu.merge.genus.mucus$Treatment <- as.factor(otu.merge.genus.mucus$Treatment)

# plot bargraphs
# BarOrder <- c("Other","Ruminococcaceae_unclassified","Roseburia","Prevotella",
# "Lachnospiraceae_unclassified","Faecalibacterium","Escherichia/Shigella",
# "Bacteroides","Alistipes")


## GENERAL PLOT
barplot.genus.mucus <-ggplot(data=otu.merge.genus.mucus, aes(x= Samplenr,
                                                             y=RelativeAbundance,
                                                             fill = factor(Category, levels=as.characte
  geom_bar(stat="identity",position = position_fill(reverse = TRUE)) +
  labs(fill = "Top 8 Genera")+scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57",
                                                           "#A660CC", "#E1A13F", "#F781BF",
                                                           "#E8D241", "#A65628","#400F73",
                                                           "#FA815F","#999999"))+
  theme_minimal()+ theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.genus.mucus
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusMucus1.ti
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.genus.mucus
dev.off()

X.order.mucus <- c(
             "Control - 0 - 0","Control - 0 - 3","Control - 0 - 5","Control - 0 - 7",".",
             "P80 - 0.05 - 0","P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 3", "SoyLec - 0.05 - 5",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - 0","SoyLec - 0.5 - 3",
             "SoyLec - 0.5 - 5","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5","RL - 0.05 - 7", ".",
             "RL - 0.5 - 0","RL - 0.5 - 3","RL - 0.5 - 5","RL - 0.5 - 7")

barplot.genus.mucus.2 <-ggplot(data=otu.merge.genus.mucus, aes(x= TreatmentTime,
                                                           y=RelativeAbundance,
                                                           fill = factor(Category, levels=as.character(o
  geom_bar(stat="identity", position = position_fill(reverse = TRUE))+
  facet_grid(Donor~., drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  labs(fill = "Top 8 Genera")+
  scale_x_discrete(limits=X.order.mucus)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
```

```
                                "#999999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.mucus.2
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusMucus2.ti
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.mucus.2
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
    barplot.genus.mucus.2
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.mucus.2
dev.off()

#P80 & RHANOLIPIDS ------------------------------------------------------------------------#

otu.merge.genus.mucus.RL.P80 <- dplyr::filter(otu.merge.genus.mucus, Emulsifier %in%
                                                c("Control","P80","RL","Fecal"))
otu.merge.genus.mucus.RL.P80$Treatment <- as.factor(otu.merge.genus.mucus.RL.P80$Treatment)

# plot bargraphs
# BarOrder <- c("Other","Ruminococcaceae_unclassified","Roseburia","Prevotella",
# "Lachnospiraceae_unclassified","Faecalibacterium","Escherichia/Shigella","Bacteroides",
# "Alistipes")

X.order <- c("Fecal - 0","Control - 0","P80 - 0.05","P80 - 0.5", "RL - 0.05","RL - 0.5")

## GENERAL PLOT
barplot.genus.mucus.RL.P80 <-ggplot(data=otu.merge.genus.mucus.RL.P80,aes(x= Samplenr,
                                                                          y=RelativeAbundance,
                                                                        fill = factor(Category, levels=as.cha
  geom_bar(stat="identity",position = position_fill(reverse = TRUE))+
  labs(fill = "Top 10 Genera")+scale_fill_manual(values =c("#CD5C5C", "#21AADE","#4EAC57",
                                                           "#A660CC", "#E1A13F", "#F781BF",
                                                           "#E8D241", "#A65628","#400F73",
```

```
                                                          "#FA815F","#999999"))+
  theme_minimal()+ theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.genus.mucus.RL.P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusMucus_RLP8
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.RL.P80
dev.off()


X.order.mucus.RL.P80 <- c(
             "Control - 0 - 0","Control - 0 - 3","Control - 0 - 5","Control - 0 - 7",".",
             "P80 - 0.05 - 0","P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5","RL - 0.05 - 7", ".",
             "RL - 0.5 - 0","RL - 0.5 - 3","RL - 0.5 - 5","RL - 0.5 - 7")


barplot.genus.mucus.RL.P80 <-ggplot(data=otu.merge.genus.mucus.RL.P80,aes(x=TreatmentTime,
                                                                          y=RelativeAbundance,
                                                                        fill = factor(Category, levels=as.cha
  geom_bar(stat="identity", position = position_fill(reverse = TRUE)) +
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.mucus.RL.P80)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
   #axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  # legend.text=element_text(size=7)
barplot.genus.mucus.RL.P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusMucusRLP8
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.RL.P80
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
   barplot.genus.mucus.RL.P80
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.mucus.RL.P80
dev.off()
```

11.1.4 25 genera

```
otu.table.t.ns.cpnr.relabun <- decostand(data.frame(otu.table.t.ns.cpnr[1:146]), 2,
                                         method = "total")
colnames(otu.table.t.ns.cpnr.relabun) <- colnames(otu.table.t.ns.cpnr)
# add taxonomy column
otu.table.t.ns.cpnr.relabun$Genus <- taxonomy.np.ns$Genus

#summate relative abundances in each sample per genus
otu.table.t.ns.cpnr.relabun.Gr <- aggregate(otu.table.t.ns.cpnr.relabun[1:146],
                                    by=list(Category=otu.table.t.ns.cpnr.relabun$Genus),
                                    FUN=sum)

#filter out 10 most abundant genusses
#take rowsums
otu.table.t.ns.cpnr.relabun.Gr$Sums <- rowSums(otu.table.t.ns.cpnr.relabun.Gr[2:147])
#order columns on value of the rowsums
otu.table.t.ns.cpnr.relabun.Gr <- otu.table.t.ns.cpnr.relabun.Gr[order(
  -otu.table.t.ns.cpnr.relabun.Gr$Sums),]
#take first 8 Genera's
otu.table.t.ns.cpnr.relabun.Gr.top25 <- otu.table.t.ns.cpnr.relabun.Gr[1:25,]
#add 11th row as a sum of all the rest
otu.table.t.ns.cpnr.relabun.Gr.top25[26,2:148] <- colSums(
  otu.table.t.ns.cpnr.relabun.Gr[26:199,2:148])

#adding "Other" on nineth place of first column
otu.table.t.ns.cpnr.relabun.Gr.top25$Category <- as.character(
  otu.table.t.ns.cpnr.relabun.Gr.top25$Category)
otu.table.t.ns.cpnr.relabun.Gr.top25$Category[26] <- c("Other")
otu.table.t.ns.cpnr.relabun.Gr.top25$Category <- as.factor(
  otu.table.t.ns.cpnr.relabun.Gr.top25$Category)

otu.table.t.ns.cpnr.relabun.Gr.top25$Sums <- NULL  #remove sums column

#rearrange data-table in order to be able to add metadata
otu.table.t.ns.cpnr.relabun.Gr.top25.R <- tidyr::gather(
```

```
  otu.table.t.ns.cpnr.relabun.Gr.top25,key = "Samplenr", value="RelativeAbundance",2:147)
otu.table.t.ns.cpnr.relabun.Gr.top25.R$Samplenr <- as.factor(
  otu.table.t.ns.cpnr.relabun.Gr.top25.R$Samplenr)

#add metadata
otu.merge.genus <- merge(otu.table.t.ns.cpnr.relabun.Gr.top25.R,metadata, by = "Samplenr")
# otu.merge.genus <- dplyr::arrange(otu.merge.genus,as.numeric(otu.merge.genus$Samplenr))
# arrange will not work to order in ascending order, since not all samplenumbers are numbers
otu.merge.genus$Samplenr <- gdata::reorder.factor(otu.merge.genus$Samplenr,
                                                  new.order=metadata$Samplenr)
#This works to arrange the dataframe in ascending order of numbers

write.xlsx(otu.merge.genus,"OTU10_Relative_Abundances.xlsx", append = TRUE,
           sheetName = "Otu_merge_genus")
```

```
otu.merge.genus.lumen <- dplyr::filter(otu.merge.genus, Location == 'lumen')
otu.merge.genus.lumen$Treatment <- as.factor(otu.merge.genus.lumen$Treatment)

# plot bargraphs
# BarOrder <- c("Other","Ruminococcaceae_unclassified","Roseburia","Prevotella",
# "Lachnospiraceae_unclassified","Faecalibacterium","Escherichia/Shigella","Bacteroides",
# "Alistipes")

X.order <- c("Fecal - 0","Control - 0","P80 - 0.05","P80 - 0.5", "SoyLec - 0.05",
             "SoyLec - 0.5","RL - 0.05","RL - 0.5")

## GENERAL PLOT
barplot.genus.lumen <-ggplot(data=otu.merge.genus.lumen, aes(x= Samplenr,
                                                             y=RelativeAbundance,
                                                             fill = factor(Category, levels=as.characte
  geom_bar(stat="identity",position = position_fill(reverse = TRUE))+
  labs(fill = "Top 8 Genera")+
scale_fill_manual(values =c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                            "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999",
                            "#CD7C5C", "#27AADE","#4EAC77", "#A670CC", "#E1A17F",
                            "#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999",
                          "#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
theme_minimal()+ theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.genus.lumen
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumen5.ti
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.genus.lumen
dev.off()


X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
```

```
            "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".", "SoyLec - 0.05 - -7",
            "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
            "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
            "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
            "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
            "RL - 0.05 - 7", ".", "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
            "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.2 <-ggplot(data=otu.merge.genus.lumen, aes(x= TreatmentTime,
                                                               y=RelativeAbundance,
                                                               fill = factor(Category, levels=as.charac
  geom_bar(stat="identity", position = position_fill(reverse = TRUE))+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999","#CD7C5C", "#27AADE","#4EAC77", "#A670CC",
                               "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73",
                               "#FA715F","#997999","#F787BF","#E8D741", "#A65728",
                               "#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.2
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusLumen6.ti
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.2
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
```

```
    barplot.genus.lumen.2
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.lumen.2
dev.off()
```

11.1.4.1 Plot Lumen

11.1.5 Faecalibacterium lumen

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
otu.merge.genus.lumen$Category <- as.factor(otu.merge.genus.lumen$Category)
otu.merge.genus.lumen.Fae <- dplyr::filter(otu.merge.genus.lumen,
                                           Category == "Faecalibacterium" )

X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".",
             "SoyLec - 0.5 - -7","SoyLec - 0.5 - 0","SoyLec - 0.5 - 1","SoyLec - 0.5 - 3",
             "SoyLec - 0.5 - 7",".","RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1",
             "RL - 0.05 - 3","RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0",
             "RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.Fae <-ggplot(data=otu.merge.genus.lumen.Fae,
                                 aes(x= TreatmentTime, y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
# scale_fill_brewer(palette="Set1")+
# scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
# "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
# scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  #"#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  #"#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  #"#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
```

```
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.lumen.Fae
dev.off()

# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
#     barplot.genus.lumen.bil
# dev.off()
#
# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
# barplot.genus.lumen.bil
# dev.off()


#P80 & RHANOLIPIDS ------------------------------------------------------------------------#
otu.merge.genus.lumen.RL.P80$Category <- as.factor(otu.merge.genus.lumen.RL.P80$Category)
otu.merge.genus.lumen.RL.P80.Fae <- dplyr::filter(otu.merge.genus.lumen.RL.P80,
                                                  Category == "Faecalibacterium" )


X.order.lumen.RL.P80 <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".", "P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".", "P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","RL - 0.05 - -7",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.RL.P80.Fae <-ggplot(data=otu.merge.genus.lumen.RL.P80.Fae,
                                        aes(x= TreatmentTime, y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.RL.P80)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
```

```
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.RL.P80.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.lumen.RL.P80.Fae
dev.off()
```

11.1.6 Faecalibacterium mucus

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
otu.merge.genus.mucus$Category <- as.factor(otu.merge.genus.mucus$Category)
otu.merge.genus.mucus.Fae <- dplyr::filter(otu.merge.genus.mucus,
                                           Category == "Faecalibacterium" )

X.order.mucus <- c( "Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
             "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5","P80 - 0.05 - 0",
             "P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".","P80 - 0.5 - -5",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "SoyLec - 0.05 - -5", "SoyLec - 0.05 - 0","SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 5", "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -5",
             "SoyLec - 0.5 - 0","SoyLec - 0.5 - 3","SoyLec - 0.5 - 5","SoyLec - 0.5 - 7",
             ".","RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".","RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")

barplot.genus.mucus.Fae <-ggplot(data=otu.merge.genus.mucus.Fae, aes(x= TreatmentTime,
                                                                     y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
```

```
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.mucus)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  #"#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  #"#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  #"#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.mucus.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.Fae
dev.off()

# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
#     barplot.genus.mucus.bil
# dev.off()
#
# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
# barplot.genus.mucus.bil
# dev.off()


#P80 & RHANOLIPIDS ------------------------------------------------------------------------#
otu.merge.genus.mucus.RL.P80$Category <- as.factor(otu.merge.genus.mucus.RL.P80$Category)
otu.merge.genus.mucus.RL.P80.Fae <- dplyr::filter(otu.merge.genus.mucus.RL.P80,
                                                  Category == "Faecalibacterium")


X.order.mucus.RL.P80 <- c("Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
              "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5",
              "P80 - 0.05 - 0","P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".",
```

```
              "P80 - 0.5 - -5","P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5",
              "P80 - 0.5 - 7",".","RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3",
              "RL - 0.05 - 5","RL - 0.05 - 7", ".","RL - 0.5 - -5","RL - 0.5 - 0",
              "RL - 0.5 - 3","RL - 0.5 - 5","RL - 0.5 - 7")

barplot.genus.mucus.RL.P80.Fae <-ggplot(data=otu.merge.genus.mucus.RL.P80.Fae,
                                        aes(x= TreatmentTime, y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.mucus.RL.P80)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # #997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.mucus.RL.P80.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.RL.P80.Fae
dev.off()
```

11.1.7 Correlation Faecalibacterium and butyrate TWeen80 and RL

```
data.SCFA.P80.0.05 <- dplyr::filter(data.SCFA,Treatment %in% c("Control - 0",
                                                               "P80 - 0.05"))
otu.merge.genus.lumen.Fae.S <- otu.merge.genus.lumen.Fae[order(
```

```
  otu.merge.genus.lumen.Fae$Samplenr),]
otu.merge.genus.lumen.Fae.P80.0.05 <- dplyr::filter(otu.merge.genus.lumen.Fae.S,
                                                    Treatment %in% c("Control - 0",
                                                                     "P80 - 0.05"))

data.SCFA.P80.0.05.D1 <- dplyr::filter(data.SCFA.P80.0.05, Donor == 1)
data.SCFA.P80.0.05.D1.F <- dplyr::filter(data.SCFA.P80.0.05.D1,Timepoint %in%
                                         c(2,11,12,14,18))
data.SCFA.P80.0.05.D2 <- dplyr::filter(data.SCFA.P80.0.05, Donor == 2)
data.SCFA.P80.0.05.D2.F <- dplyr::filter(data.SCFA.P80.0.05.D2,Timepoint %in%
                                         c(2,9,10,12,16))

data.SCFA.P80.0.05.F <- rbind(data.SCFA.P80.0.05.D2.F,data.SCFA.P80.0.05.D1.F)

Corr_Fae_Lumen_0.05 <- cor(data.SCFA.P80.0.05.F$Butyric_acid,
                           otu.merge.genus.lumen.Fae.P80.0.05$RelativeAbundance,
                           method = c("spearman"))  # 0.5068
Corr_Fae_Lumen_0.05_sign <- cor.test(data.SCFA.P80.0.05.F$Butyric_acid,
                                     otu.merge.genus.lumen.Fae.P80.0.05$RelativeAbundance,
                                     method=c("spearman"))
  # 0.5068
  #p = 0.0241

data.SCFA.P80.0.5 <- dplyr::filter(data.SCFA,Treatment %in% c("Control - 0","P80 - 0.5"))
otu.merge.genus.lumen.Fae.S <- otu.merge.genus.lumen.Fae[order(
  otu.merge.genus.lumen.Fae$Samplenr),]
otu.merge.genus.lumen.Fae.P80.0.5 <- dplyr::filter(otu.merge.genus.lumen.Fae.S,
                                                   Treatment %in% c("Control - 0",
                                                                    "P80 - 0.5"))

data.SCFA.P80.0.5.D1 <- dplyr::filter(data.SCFA.P80.0.5, Donor == 1)
data.SCFA.P80.0.5.D1.F <- dplyr::filter(data.SCFA.P80.0.5.D1,Timepoint %in%
                                         c(2,11,12,14,18))
data.SCFA.P80.0.5.D2 <- dplyr::filter(data.SCFA.P80.0.5, Donor == 2)
data.SCFA.P80.0.5.D2.F <- dplyr::filter(data.SCFA.P80.0.5.D2,Timepoint %in%
                                         c(2,9,10,12,16))

data.SCFA.P80.0.5.F <- rbind(data.SCFA.P80.0.5.D2.F,data.SCFA.P80.0.5.D1.F)

Corr_Fae_Lumen_0.5 <- cor(data.SCFA.P80.0.5.F$Butyric_acid,
                          otu.merge.genus.lumen.Fae.P80.0.5$RelativeAbundance,
                          method = c( "spearman")) # 0.6407
Corr_Fae_Lumen_0.5_sign <- cor.test(data.SCFA.P80.0.5.F$Butyric_acid,
                                    otu.merge.genus.lumen.Fae.P80.0.5$RelativeAbundance,
                                    method=c("spearman"))
  # 0.6407
  # p = 0.003


data.SCFA.RL.0.05 <- dplyr::filter(data.SCFA,Treatment %in% c("Control - 0","RL - 0.05"))
otu.merge.genus.lumen.Fae.S <- otu.merge.genus.lumen.Fae[order(
  otu.merge.genus.lumen.Fae$Samplenr),]
otu.merge.genus.lumen.Fae.RL.0.05 <- dplyr::filter(otu.merge.genus.lumen.Fae.S,
```

```
                                                    Treatment %in% c("Control - 0",
                                                                     "RL - 0.05"))

data.SCFA.RL.0.05.D1 <- dplyr::filter(data.SCFA.RL.0.05, Donor == 1)
data.SCFA.RL.0.05.D1.F <- dplyr::filter(data.SCFA.RL.0.05.D1,Timepoint %in%
                                          c(2,11,12,14,18))
data.SCFA.RL.0.05.D2 <- dplyr::filter(data.SCFA.RL.0.05, Donor == 2)
data.SCFA.RL.0.05.D2.F <- dplyr::filter(data.SCFA.RL.0.05.D2,Timepoint %in%
                                          c(2,9,10,12,16))

data.SCFA.RL.0.05.F <- rbind(data.SCFA.RL.0.05.D2.F,data.SCFA.RL.0.05.D1.F)

Corr_Fae_Lumen_0.05 <- cor(data.SCFA.RL.0.05.F$Butyric_acid,
                           otu.merge.genus.lumen.Fae.RL.0.05$RelativeAbundance,
                           method = c("spearman"))  # 0.7682
Corr_Fae_Lumen_0.05_sign <- cor.test(data.SCFA.RL.0.05.F$Butyric_acid,
                                     otu.merge.genus.lumen.Fae.RL.0.05$RelativeAbundance,
                                     method=c("spearman"), exact = FALSE)
  # 0.7682
  #p = 7.629e-5

data.SCFA.RL.0.5 <- dplyr::filter(data.SCFA,Treatment %in% c("Control - 0","RL - 0.5"))
otu.merge.genus.lumen.Fae.S <- otu.merge.genus.lumen.Fae[order(
  otu.merge.genus.lumen.Fae$Samplenr),]
otu.merge.genus.lumen.Fae.RL.0.5 <- dplyr::filter(otu.merge.genus.lumen.Fae.S,
                                                  Treatment %in% c("Control - 0",
                                                                   "RL - 0.5"))

data.SCFA.RL.0.5.D1 <- dplyr::filter(data.SCFA.RL.0.5, Donor == 1)
data.SCFA.RL.0.5.D1.F <- dplyr::filter(data.SCFA.RL.0.5.D1,Timepoint %in%
                                         c(2,11,12,14,18))
data.SCFA.RL.0.5.D2 <- dplyr::filter(data.SCFA.RL.0.5, Donor == 2)
data.SCFA.RL.0.5.D2.F <- dplyr::filter(data.SCFA.RL.0.5.D2,Timepoint %in%
                                         c(2,9,10,12,16))

data.SCFA.RL.0.5.F <- rbind(data.SCFA.RL.0.5.D2.F,data.SCFA.RL.0.5.D1.F)

Corr_Fae_Lumen_0.5 <- cor(data.SCFA.RL.0.5.F$Butyric_acid,
                          otu.merge.genus.lumen.Fae.RL.0.5$RelativeAbundance,
                          method = c( "spearman")) # 0.7398753
Corr_Fae_Lumen_0.5_sign <- cor.test(data.SCFA.RL.0.5.F$Butyric_acid,
                                    otu.merge.genus.lumen.Fae.RL.0.5$RelativeAbundance,
                                    method=c("spearman"),exact = FALSE)
  # 0.7399
  #p = 0.000192
```

11.1.8 Bilophyla lumen

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
otu.merge.genus.lumen$Category <- as.factor(otu.merge.genus.lumen$Category)
otu.merge.genus.lumen.Bil <- dplyr::filter(otu.merge.genus.lumen,Category == "Bilophila")
```

```
X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.bil <-ggplot(data=otu.merge.genus.lumen.Bil,aes(x= TreatmentTime,
                                                                    y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.bil
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusBilophila
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.lumen.bil
dev.off()
```

```
pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
    barplot.genus.lumen.bil
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.lumen.bil
dev.off()


#P80 & RHANOLIPIDS ------------------------------------------------------------------------#
otu.merge.genus.lumen.RL.P80$Category <- as.factor(otu.merge.genus.lumen.RL.P80$Category)
otu.merge.genus.lumen.RL.P80.Bil <- dplyr::filter(otu.merge.genus.lumen.RL.P80,
                                                  Category == "Bilophila" )


X.order.lumen.RL.P80 <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".", "P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","RL - 0.05 - -7",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.RL.P80.Bil <-ggplot(data=otu.merge.genus.lumen.RL.P80.Bil,
                                        aes(x= TreatmentTime, y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.RL.P80)+
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
```

```
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.RL.P80.Bil
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusBilophila
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.lumen.RL.P80.Bil
dev.off()
```

11.1.9 Bilophyla mucus

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
otu.merge.genus.mucus$Category <- as.factor(otu.merge.genus.mucus$Category)
otu.merge.genus.mucus.Bil <- dplyr::filter(otu.merge.genus.mucus,
                                           Category == "Bilophila")

X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".", "P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.mucus.bil <-ggplot(data=otu.merge.genus.mucus.Bil,aes(x= TreatmentTime,
                                                                    y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.mucus)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
```

```
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.mucus.bil
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusBilophila
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.bil
dev.off()

# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
#     barplot.genus.mucus.bil
# dev.off()
#
# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
# barplot.genus.mucus.bil
# dev.off()


#P80 & RHANOLIPIDS ------------------------------------------------------------------------#
otu.merge.genus.mucus.RL.P80$Category <- as.factor(otu.merge.genus.mucus.RL.P80$Category)
otu.merge.genus.mucus.RL.P80.Bil <- dplyr::filter(otu.merge.genus.mucus.RL.P80,
                                                  Category == "Bilophila" )


X.order.mucus.RL.P80 <-c("Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
             "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5", "P80 - 0.05 - 0",
             "P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".","P80 - 0.5 - -5",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".", "RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")

barplot.genus.mucus.RL.P80.Bil <-ggplot(data=otu.merge.genus.mucus.RL.P80.Bil,
                                        aes(x= TreatmentTime, y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.mucus.RL.P80)+
```

```
  # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
# theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
# theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.mucus.RL.P80.Bil
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusBilophila
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.mucus.RL.P80.Bil
dev.off()
```

11.1.10 Roseburia lumen

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
otu.merge.genus.lumen$Category <- as.factor(otu.merge.genus.lumen$Category)
otu.merge.genus.lumen.Butyr <- dplyr::filter(otu.merge.genus.lumen,
                                             Category == "Roseburia")

X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".", "SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".", "SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.bil <-ggplot(data=otu.merge.genus.lumen.Bil,aes(x= TreatmentTime,
```

```
                                                                y=RelativeAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  # "#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  # "#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  # "#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  # "#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
# legend.text=element_text(size=7))
barplot.genus.lumen.bil
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusBilophila
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.bil
dev.off()

pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
    barplot.genus.lumen.bil
dev.off()

png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
barplot.genus.lumen.bil
dev.off()
```

12 Alpha diversity

12.1 Chao1

```
expshared <- ceiling(otu.table.t.ns.cpnr)
#expshared <- expshared[-grep("N",expshared$Emulsifier),]

calcspecies <- function(x)
{
  #x are the counts of an individual sample
  #some consistency checks
  if(length(x!=0)>10) # the sample has to have at least 10 species
  {
    if(sum(x[which(x!=0)])>length(x[which(x!=0)])) #there has to be at least some
      # variation and not just all 0 counts
    {
    inputdf <- data.frame(j=as.numeric(as.character(rownames(as.matrix(table(x))))),
                          n_j=as.vector(table(x)))
    inputdf <- subset(inputdf,subset=inputdf$j!=0)

    #estimators don't work with 0 observations
    ##following is the calculation of diversity parameters
    chaores <- chao1984(inputdf)
    suppressWarnings(chaobung <- SPECIES::ChaoBunge(inputdf,t=10))
    suppressWarnings(ace1res <- SPECIES::ChaoLee1992(inputdf,t=10,method="ACE-1"))


    #pcgres <- SPECIES::pcg(inputdf) #too computationally intensive , overparametrized
    #unplmeres <- SPECIES::unpmle(inputdf,C=1,b=20,seed=456789,dis=0)
    #quite computationally intensive, certainly for larger datasets
    #pnplmeres <- SPECIES::pnpmle(inputdf,C=1,b=20,seed=456789,dis=0)
    #quite computationally intensive, certainly for larger datasets
    #jacknres <- invisible(suppressWarnings(suppressMessages(SPECIES::jackknife(inputdf))))
    #could not suppress output

    resdf <- data.frame(Methods=c("Chao1984","ChaoBunge2002","ACE-1"),
                        Nhat=c(chaores$Nhat,chaobung$Nhat,ace1res$Nhat),
                        SE_low=c(chaores$Nhat-chaores$SE,chaobung$Nhat-chaobung$SE,
                                 ace1res$Nhat-ace1res$SE),
                        SE_high=c(chaores$Nhat+chaores$SE,chaobung$Nhat+chaobung$SE,
                                  ace1res$Nhat+ace1res$SE),
                        CI_lb=c(chaores$CI[1],chaobung$CI[1],ace1res$CI[1]),
                        CI_ub=c(chaores$CI[2],chaobung$CI[2],ace1res$CI[2]))
    }else{
      resdf <- data.frame(Methods=c("Chao1984","ChaoBunge2002","ACE-1"),
                        Nhat=c(NA,NA,NA),
                        SE_low=c(NA,NA,NA),
                        SE_high=c(NA,NA,NA),
                        CI_lb=c(NA,NA,NA),
                        CI_ub=c(NA,NA,NA))
    }

  }else{
```

```
    resdf <- data.frame(Methods=c("Chao1984","ChaoBunge2002","ACE-1"),
                        Nhat=c(NA,NA,NA),
                        SE_low=c(NA,NA,NA),
                        SE_high=c(NA,NA,NA),
                        CI_lb=c(NA,NA,NA),
                        CI_ub=c(NA,NA,NA))
  }
  return(resdf)
}

# Below we list the table with several richness estimators (per sample) and their
# confidence intervals.
fullsetrichcalc <- mclapply(expshared,calcspecies,mc.cores = 12,mc.cleanup = TRUE)
tablrichdf <- ldply(fullsetrichcalc,function(x)x)
names(tablrichdf)[1]<-"Samplenr"
kable(tablrichdf,caption = "Richness estimators with standard errors and 95\\%
      confidence interval")
write.csv(tablrichdf,"SHIME_Richness_estimated.csv")

### Processing of Richness parameters
tablrichdf

rich.merge <- merge(tablrichdf,metadata, by = "Samplenr")
rich.merge.chao <- dplyr::filter(rich.merge, Methods == "Chao1984")

rich.merge.chao.lumen <- dplyr::filter(rich.merge.chao, Location == "lumen")
rich.merge.chao.lumen$Samplenr <- gdata::reorder.factor(rich.merge.chao.lumen$Samplenr,
                                                        new.order=metadata$Samplenr)
rich.merge.chao.mucus <- dplyr::filter(rich.merge.chao, Location == "mucin")
rich.merge.chao.mucus$Samplenr <- gdata::reorder.factor(rich.merge.chao.mucus$Samplenr,
                                                        new.order=metadata$Samplenr)


X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".", "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")


EM_Colors <- c("#CD5C5C", "#999999","#4EAC57", "#21AADE", "#A660CC", "#E1A13F")

## PLOT Chao1 Parameter
barplot.chao.lumen <-ggplot(data=rich.merge.chao.lumen,aes(x= TreatmentTime, y=Nhat,
                                                        fill = as.factor(Emulsifier))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
```

```
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  #scale_fill_brewer(palette="Set1")+
  scale_fill_manual(values = EM_Colors)+
  ggtitle("Chao richness")+
  ylab("Chao")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.chao.lumen
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotCHAOLumen1.tif
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.chao.lumen
dev.off()

png('Plot_Batch_Illumina_Run3_Chao1richness.png', res = 300, width = 3500, height = 1500)
  barplot.chao.lumen
dev.off()


X.order.mucus <- c("Fecal - 0 - 0",".",
             "Control - 0 - -5","Control - 0 - 0","Control - 0 - 3","Control - 0 - 5",
             "Control - 0 - 7",".", "P80 - 0.05 - -5", "P80 - 0.05 - 0","P80 - 0.05 - 3",
             "P80 - 0.05 - 5","P80 - 0.05 - 7",".","P80 - 0.5 - -5","P80 - 0.5 - 0",
             "P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".","SoyLec - 0.05 - -5",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 3", "SoyLec - 0.05 - 5",
             "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - -5","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 3","SoyLec - 0.5 - 5","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".","RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")

barplot.chao.mucus <-ggplot(data=rich.merge.chao.mucus,aes(x= TreatmentTime, y=Nhat,
                                                           fill = as.factor(Emulsifier))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  # legend.text=element_text(size=7))
```

```
  scale_fill_brewer(palette="Set1")+
  ggtitle("Chao richness")+
  ylab("Chao")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.chao.mucus
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotCHAOMucus1.tif
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
barplot.chao.mucus
dev.off()


#-------------------------------------------------------------------------------------------#
#RL - P80
rich.merge.chao.lumen.P80.RL <- dplyr::filter(rich.merge.chao.lumen,Emulsifier %in%
                                                c("Fecal","Control","RL","P80"))
rich.merge.chao.mucus.P80.RL <- dplyr::filter(rich.merge.chao.mucus,Emulsifier %in%
                                                c("Fecal","Control","RL","P80"))

X.order.lumen.P80.RL <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".", "P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".", "RL - 0.05 - -7",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")


EM_Colors <- c("#4a454b", "#999999","#d76ae5", "#873292","Lightblue", "DarkBlue",
               "#A660CC", "#E1A13F")

## PLOT Chao1 Parameter
barplot.chao.lumen.P80.RL <-ggplot(data=rich.merge.chao.lumen.P80.RL,aes(x= TreatmentTime,
                                                                         y=Nhat,
                                                                fill = as.factor(Treatment))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  # theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
```

```
  # legend.text=element_text(size=7))
  # scale_fill_brewer(palette="Set1")+
  scale_fill_manual(values = EM_Colors)+
  ggtitle("Chao richness")+
  ylab("Chao")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen.P80.RL)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.chao.lumen.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotCHAOLumen1_P80_
     width=10*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.chao.lumen.P80.RL
dev.off()

#png('Plot_Batch_Illumina_Run3_Chao1richness.png', res = 300, width = 3500, height = 1500)
 # barplot.chao.lumen.Soy
#dev.off()


X.order.mucus.P80.RL <- c("Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
             "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5", "P80 - 0.05 - 0",
             "P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".", "P80 - 0.5 - -5",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".", "RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")

EM_Colors_mucus <- c("#4a454b", "#d76ae5", "#873292","Lightblue", "DarkBlue", "#A660CC",
                     "#E1A13F")

barplot.chao.mucus.P80.RL <-ggplot(data=rich.merge.chao.mucus.P80.RL, aes(x= TreatmentTime,
                                                                          y=Nhat,
                                                                fill = as.factor(Treatment))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  # legend.text=element_text(size=7))
  scale_fill_manual(values = EM_Colors_mucus)+
  ggtitle("Chao richness")+
```

```
  ylab("Chao")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus.P80.RL)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.chao.mucus.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotCHAOMucus1_P80_
     width=10*dpi,height=6*dpi,
     res=dpi,compression="lzw")
barplot.chao.mucus.P80.RL
dev.off()
```

12.2 Shannon, Simpson

```
expshared <- ceiling(otu.table.t.ns.cpnr)
#expshared <- expshared[-grep("N",expshared$Emulsifier),]

csumssto <- colSums(expshared)  # sums of reads per OTU
vegplotsto <- data.frame(numSampled=rep(1,ncol(expshared)),Sample=colnames(expshared),
                         Shannon=0,Invsimpson=1,Simpson=0,Pielou=0)
# creates matrix where parameters will be filled in

#Calculating and filling in the 4 parameters into the vegplotsto
for(i in seq(100,max(colSums(expshared)),by=round(max(colSums(expshared))/100L)))
{
  colselsto <- csumssto >= i
  ipraresto <- expshared[,colselsto]
  raredsto <- rrarefy(t(ipraresto),sample=i)
  shansto <- vegan::diversity(raredsto,index="shannon")
  simpsto <- vegan::diversity(raredsto,index="simpson")
  invsimpsto <- vegan::diversity(raredsto,index="invsimpson")
  pielousto <- vegan::diversity(raredsto,index="shannon")/specnumber(raredsto)
  if(is.null(colnames(ipraresto)))
  {
    ipraresto<-data.frame(col1<-ipraresto)
    colnames(ipraresto)<-names(csumssto[csumssto==max(csumssto)])
  }
  vegplotsto <- rbind(vegplotsto,data.frame(numSampled=i,Sample=colnames(ipraresto),
                                            Shannon=as.vector(shansto),
```

```
                                        Invsimpson=as.vector(invsimpsto),
                                        Simpson=as.vector(simpsto),
                                        Pielou=pielousto))
}

#-------------------------------------------------------------------------------#
# vegplotsto$Sample <- mapvalues(vegplotsto$Sample,
#                                from=levels(vegplotsto$Sample),
#                                to=sub("(...).*","\\1",levels(vegplotsto$Sample)))

# shanplotveg <- ggplot(aes(x=numSampled,y=Shannon,color=Sample),data=vegplotsto) +
# geom_point(alpha=0.6) + ggtitle("Batch experiments - Shannon diversity index")
# shanplotveg + geom_line() + ylab("Shannon diversity index (H)") +
#xlab("Reads sampled") + theme(legend.position = "none")
# spexp <- shanplotveg + geom_line() + ylab("Shannon diversity index (H)") +
#xlab("Reads sampled")+ theme(legend.position = "none")
#
# highqualgraphR(spexp,filename = "Shannon_collector",extension = c("png"))
#
# simpplotveg <- ggplot(aes(x=numSampled,y=Simpson,color=Sample),data=vegplotsto) +
#geom_point(alpha=0.6) + ggtitle("Batch experiments - Simpson diversity index")
# simpplotveg + geom_line() + ylab("Simpson diversity index (1-D)") +
#xlab("Reads sampled")+ theme(legend.position = "none")
# sipexp <- simpplotveg + geom_line() + ylab("Simpson diversity index (1-D)") +
#xlab("Reads sampled")+ theme(legend.position = "none")
#
# highqualgraphR(sipexp,filename = "Simpson_collector",extension = c("png"))
#
# invsimpplotveg <- ggplot(aes(x=numSampled,y=Invsimpson,color=Sample),
#data=vegplotsto) +
#geom_point(alpha=0.6)+ ggtitle("Batch experiments - Inv. Simpson diversity index")+
#theme(legend.position = "none")
# invsimpplotveg + geom_line() + ylab("Inverse Simpson diversity index (1/D)") +
#xlab("Reads sampled")
# invsiexp <- invsimpplotveg + geom_line() +
#ylab("Inverse Simpson diversity index (1/D)") + xlab("Reads sampled")
#
#
# highqualgraphR(invsiexp,filename = "Inverse_Simpson_collector",
#extension = c("png"))
#
# pielouplotveg <- ggplot(aes(x=numSampled,y=Pielou,color=Sample),data=vegplotsto) +
#geom_point(alpha=0.6)+ ggtitle("Batch experiments - Inv. Simpson diversity index")+
#theme(legend.position = "none")
# pielouplotveg + geom_line() + ylab("Pielou evenness index (J)") + xlab("Reads sampled")
# pieexp <- pielouplotveg + geom_line() + ylab("Pielou evenness index (J)") +
#xlab("Reads sampled")
#
# highqualgraphR(pieexp,filename = "Pielou_collector",extension = c("png"))

#The diversity estimators which are stable (i.e. the plots level off) should be considered
#for further evaluation. Their values on the full dataset (without singletons) are
#listed in the table below.
```

```
#-------------------------------------------------------------------------------------------#
vegplotdf <- vegplotsto
#vegplotdf$Sample <- plyr::mapvalues(vegplotdf$Sample,
#                                    from = levels(vegplotdf$Sample),
#                                    names(shared.t.ns))
vegtable <- vegplotdf  %>%  group_by(Sample)  %>%
  dplyr::filter(numSampled==max(numSampled)) %>%
                dplyr::filter(numSampled!=1)
#%>%  arrange(Sample), leave arranged by numSampled
#if numSampled == 1 then there was a prob with the sample size < 100 reads
kable(vegtable,caption="Diversity indices on the full dataset")
write.csv(vegtable,"Diversity_vegan.csv")

#-------------------------------------------------------------------------------------------#
#rarefaciton type curve
Shannonplot <- ggplot(data = vegplotdf, aes(x=numSampled, y= Shannon, color= Sample))+
  geom_point()+ theme_minimal()+xlab("numSampled")+ylab("Shannon index")
Shannonplot
```

```
#-------------------------------------------------------------------------------------------#

colnames(vegtable)[2] <- "Samplenr"
vegtable.merge <- merge(vegtable,metadata, by = "Samplenr")
vegtable.merge.lumen <- dplyr::filter(vegtable.merge,Location=="lumen")
vegtable.merge.lumen$Samplenr <- gdata::reorder.factor(vegtable.merge.lumen$Samplenr,
                                                       new.order=metadata$Samplenr)
vegtable.merge.mucus <- dplyr::filter(vegtable.merge,Location=="mucin")
vegtable.merge.mucus$Samplenr <- gdata::reorder.factor(vegtable.merge.mucus$Samplenr,
                                                       new.order=metadata$Samplenr)


Barplot.Shannon.lumen <- ggplot(data = vegtable.merge.lumen, aes(x=TreatmentTime,
                                                                 y=Shannon,
                                                                fill=as.factor(Emulsifier)))+
  geom_bar(stat="identity", position = "dodge")+facet_grid(.~Donor)+ theme_minimal()+
  xlab("numSampled")+ylab("Shannon index")+theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  scale_fill_brewer(palette="Set1")+
  ggtitle("Shannon")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
```

```
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
Barplot.Shannon.lumen
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSHANNONLumen1.t
     width=10*dpi,height=6*dpi,res=dpi,compression="lzw")
Barplot.Shannon.lumen
dev.off()


Barplot.Shannon.mucus <- ggplot(data = vegtable.merge.mucus, aes(x=TreatmentTime,
                                                                 y=Shannon,
                                                                fill=as.factor(Emulsifier)))+
  geom_bar(stat="identity", position = "dodge")+facet_grid(.~SHIME)+ theme_minimal()+
  xlab("numSampled")+ylab("Shannon index")+theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  scale_fill_brewer(palette="Set1")+
  ggtitle("Shannon")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
Barplot.Shannon.mucus
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSHANNONMucus1.t
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
Barplot.Shannon.mucus
dev.off()


Barplot.Simpson.lumen <- ggplot(data = vegtable.merge.lumen, aes(x=TreatmentTime, y=Simpson, fill=as.fa
  facet_grid(.~SHIME)+ theme_minimal()+xlab("numSampled")+ylab("Simpson index")+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  scale_fill_brewer(palette="Set1")+
  ggtitle("Simpson")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
```

```
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
Barplot.Simpson.lumen
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSIMPSONLumen1.t
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
Barplot.Simpson.lumen
dev.off()
```

```
Barplot.Simpson.mucus <- ggplot(data = vegtable.merge.mucus, aes(x=TreatmentTime,
                                                                 y=Simpson,
                                                                fill=as.factor(Emulsifier)))+
  geom_bar(stat="identity", position = "dodge")+facet_grid(.~SHIME)+ theme_minimal()+
  xlab("numSampled")+ylab("Simpson index")+theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  scale_fill_brewer(palette="Set1")+
  ggtitle("Simpson")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 10),
        axis.text.y=element_text(size = 6))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus)+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
Barplot.Simpson.mucus
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSIMPSONMucus1.t
     width=12*dpi,height=8*dpi,res=dpi,compression="lzw")
Barplot.Simpson.mucus
dev.off()
```

```
#-------------------------------------------------------------------------------------#
#RL - P80
vegtable.merge.lumen.P80.RL <- dplyr::filter(vegtable.merge.lumen,Emulsifier %in%
                                               c("Fecal","Control","RL","P80"))
vegtable.merge.mucus.P80.RL <- dplyr::filter(vegtable.merge.mucus,Emulsifier %in%
                                               c("Fecal","Control","RL","P80"))

X.order.lumen.P80.RL <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".","RL - 0.05 - -7",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")


EM_Colors <- c("#4a454b", "#999999","#d76ae5", "#873292","Lightblue", "DarkBlue",
               "#A660CC", "#E1A13F")

## PLOT Chao1 Parameter
barplot.Shannon.lumen.P80.RL <-ggplot(data=vegtable.merge.lumen.P80.RL,
                                      aes(x= TreatmentTime, y=Shannon,
                                          fill = as.factor(Treatment))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  #scale_fill_brewer(palette="Set1")+
  scale_fill_manual(values = EM_Colors)+
  #ggtitle("Shannon richness")+
  ylab("Shannon")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 12))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen.P80.RL)+
  theme(axis.text.y = element_text(size = 12,family='sans',angle=0),
        axis.title.x = element_text(size =16,family='sans',angle=0),
        axis.title.y = element_text(size=16,family='sans',angle=90),
        legend.text = element_text(size=14,family='sans',angle=0),
        legend.title = element_text(size=16,family='sans',angle=0),
        strip.text = element_text(size=14,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.Shannon.lumen.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSHANNONLumen1_
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.Shannon.lumen.P80.RL
```

```
dev.off()

#png('Plot_Batch_Illumina_Run3_Chao1richness.png', res = 300, width = 3500, height = 1500)
 # barplot.chao.lumen.Soy
#dev.off()


X.order.mucus.P80.RL <- c("Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
             "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5", "P80 - 0.05 - 0",
             "P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".","P80 - 0.5 - -5",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".","RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")

EM_Colors_mucus <- c("#4a454b", "#d76ae5", "#873292", "Lightblue", "DarkBlue",
                     "#A660CC", "#E1A13F")

barplot.Shannon.mucus.P80.RL <-ggplot(data=vegtable.merge.mucus.P80.RL,
                                      aes(x= TreatmentTime, y=Shannon,
                                          fill = as.factor(Treatment))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  # legend.text=element_text(size=7))
  scale_fill_manual(values = EM_Colors_mucus)+
  #ggtitle("Shannon richness")+
  ylab("Shannon")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 12))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus.P80.RL)+
  theme(axis.text.y = element_text(size = 12,family='sans',angle=0),
        axis.title.x = element_text(size =16,family='sans',angle=0),
        axis.title.y = element_text(size=16,family='sans',angle=90),
        legend.text = element_text(size=14,family='sans',angle=0),
        legend.title = element_text(size=16,family='sans',angle=0),
        strip.text = element_text(size=14,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.Shannon.mucus.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSHANNONMucus1_I
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.Shannon.mucus.P80.RL
dev.off()

## PLOT Simpson Parameter
barplot.Simpson.lumen.P80.RL <-ggplot(data=vegtable.merge.lumen.P80.RL,
```

```
                                    aes(x= TreatmentTime, y=Simpson,
                                        fill = as.factor(Treatment))) +
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  #scale_fill_brewer(palette="Set1")+
  scale_fill_manual(values = EM_Colors)+
  #ggtitle("Simpson richness")+
  ylab("Simpson")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 12))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.lumen.P80.RL)+
  theme(axis.text.y = element_text(size = 12,family='sans',angle=0),
        axis.title.x = element_text(size =16,family='sans',angle=0),
        axis.title.y = element_text(size=16,family='sans',angle=90),
        legend.text = element_text(size=14,family='sans',angle=0),
        legend.title = element_text(size=14,family='sans',angle=0),
        strip.text = element_text(size=14,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.Simpson.lumen.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSIMPSONLumen1_I
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.Simpson.lumen.P80.RL
dev.off()

#png('Plot_Batch_Illumina_Run3_Chao1richness.png', res = 300, width = 3500, height = 1500)
 # barplot.chao.lumen.Soy
#dev.off()


X.order.mucus.P80.RL <- c("Control - 0 - -5","Control - 0 - 0","Control - 0 - 3",
             "Control - 0 - 5","Control - 0 - 7",".","P80 - 0.05 - -5", "P80 - 0.05 - 0",
             "P80 - 0.05 - 3","P80 - 0.05 - 5","P80 - 0.05 - 7",".","P80 - 0.5 - -5",
             "P80 - 0.5 - 0","P80 - 0.5 - 3","P80 - 0.5 - 5","P80 - 0.5 - 7",".",
             "RL - 0.05 - -5","RL - 0.05 - 0","RL - 0.05 - 3","RL - 0.05 - 5",
             "RL - 0.05 - 7", ".","RL - 0.5 - -5","RL - 0.5 - 0","RL - 0.5 - 3",
             "RL - 0.5 - 5","RL - 0.5 - 7")


EM_Colors_mucus <- c("#4a454b", "#d76ae5", "#873292","Lightblue", "DarkBlue", "#A660CC",
                     "#E1A13F")

barplot.Simpson.mucus.P80.RL <-ggplot(data=vegtable.merge.mucus.P80.RL,
                                    aes(x= TreatmentTime, y=Simpson,
                                        fill = as.factor(Treatment))) +
```

```
  geom_bar(stat="identity", position = "dodge")+
  facet_grid(~Donor, drop = TRUE)+
  theme_minimal()+
  #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
  #legend.text=element_text(size=7))
  scale_fill_manual(values = EM_Colors_mucus)+
  #ggtitle("Simpson richness")+
  ylab("Simpson")+
  theme(axis.text.x=element_text(angle=90, hjust=1, size = 12))+
  labs(fill = "Emulsifier")+
  scale_x_discrete(limits=X.order.mucus.P80.RL)+
  theme(axis.text.y = element_text(size = 12,family='sans',angle=0),
        axis.title.x = element_text(size =16,family='sans',angle=0),
        axis.title.y = element_text(size=16,family='sans',angle=90),
        legend.text = element_text(size=14,family='sans',angle=0),
        legend.title = element_text(size=16,family='sans',angle=0),
        strip.text = element_text(size=14,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
barplot.Simpson.mucus.P80.RL
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotSIMPSONMucus1_
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.Simpson.mucus.P80.RL
dev.off()

#-------------------------------------------------------------------------------------#
EM_Colors <- c("#F4685CFF", "#AB337CFF","#FDE4A6FF","#681D81FF","#FEB37BFF","#D6456CFF")
#                CMC           Control    tween80    RL          SL          SoyLec

#Plot parameters separately with their significance tests: SHANNON
EM_Colors <- c("#CD5C5C", "#4EAC57", "#21AADE", "#A660CC", "#E1A13F", "#999999")
```

13 Ordination (NMDS; measure of beta diversity or between sample diversity)

13.1 PCOA

```
# RelAbs <- decostand(data.frame(otu.table.t.ns.cpnr[1:142]), 2, method = "total")
RelAbs <- otu.table.t.ns.cpnr.relabun[1:142]

#create distance matrix of relative abundances of OTU-table
disjac <- vegan::vegdist(t(RelAbs), method="jaccard") #computes dissimilarity indices
# disjac.cs <- vegdist(t(shared.cs), method="jaccard")

# PCOA.Rel<- pcoa(disjac, correction="none", rn=NULL)
```

```
PCOA.Rel<- prcomp(disjac, center = TRUE, scale = TRUE)  #==> normalize or not????
PCOA.Rel<- prcomp(disjac)

# S3 method for pcoa
# biplot(PCOA.Rel, Y=NULL, plot.axes = c(1,2), dir.axis1=1,
#        dir.axis2=1, rn=NULL, main=NULL)
# #plot using ggbiplot
# Rel.biplot <- ggbiplot(PCOA.Rel, obs.scale = 1, var.scale = 1,
#                        groups = metadata$SHIME[1:142],
#               ellipse = TRUE,
#               circle = TRUE)
#
# plot(PCOA.Rel)
# summary(PCOA.Rel)

#plot using ggplot
PCOA.Rel.data <- as.data.frame(PCOA.Rel$rotation)[1:2]
PCOA.Rel.data$Samplenr <- rownames(PCOA.Rel.data)
PCOA.Rel.merge <- merge(PCOA.Rel.data,metadata,by = "Samplenr")

PCOA.Rel.plot1 <- ggplot(PCOA.Rel.merge, aes(x = PC1, y = PC2,
                                             color = as.factor(Location),
                                             shape = as.factor(Donor))) +
  geom_point(size = 3) + theme_bw() + labs(color = "Location",shape ="Donor")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
  scale_color_manual(values = c("#754ba6","#faed75","mediumblue", "orange"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
  # theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
PCOA.Rel.plot1
```

```
PCOA.Rel.plot2 <- ggplot(PCOA.Rel.merge, aes(x = PC1, y = PC2,
                                             color = as.factor(Treatment),
                                             shape = as.factor(Donor))) +
  geom_point(size = 3) + theme_bw() + labs(color = "Treatment",shape ="Donor")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
```

```
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot2
```

```
PCOA.Rel.plot3 <- ggplot(PCOA.Rel.merge, aes(x = PC1, y = PC2,
                                             color = as.factor(Timepoint),
                                             shape = as.factor(Donor))) +
  geom_point(size = 3) + theme_bw() + labs(color = "Timepoint",shape ="Donor")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  scale_color_viridis(option="D",discrete = TRUE)+
  # scale_color_manual(values = c("#0c8560","#999999", "#d28eed", "#844da8" ,"#f2b385",
  #"#d97914", "#f5e2a2", "#e3d029", "#F781BF","#74dff7", "#125087"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot3
```

```
PCOA.Rel.plot4 <- ggplot(PCOA.Rel.merge, aes(x = PC1, y = PC2,
                                             color = as.factor(Treatment))) +
  geom_point(aes(size = Timepoint)) + geom_text(aes(x = PC1,y= PC2, label = Timepoint),
                                                hjust = -1, vjust = -1)+
  facet_grid(SHIME~Location)+ theme_bw() + labs(color = "Treatment")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="D",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot4
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA1.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot1
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA2.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot2
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA3.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot3
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA4.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot4
dev.off()

PCOA.Rel.merge.lumen <- dplyr::filter(PCOA.Rel.merge,Location == "lumen")

PCOA.Rel.plot.lumen <- ggplot(PCOA.Rel.merge.lumen, aes(x = PC1, y = PC2,
                                                        color = as.factor(Treatment),
                                                    shape = as.factor(SHIME), size = 4)) +
  geom_point()  + geom_text(aes(x = PC1, y = PC2,label= Timepoint),hjust = 0,
                            nudge_y = 0.1, color = "Gray") + theme_minimal()
PCOA.Rel.plot.lumen
```

```
PCOA.Rel.merge.mucus <- dplyr::filter(PCOA.Rel.merge,Location == "mucin")
PCOA.Rel.plot.mucus <- ggplot(PCOA.Rel.merge.mucus,
                              aes(x = PC1, y = PC2, color = as.factor(Treatment),
                                  shape = as.factor(SHIME), size = 4)) +
  geom_point()  + geom_text(aes(x = PC1, y = PC2,label= Timepoint),hjust = 0,
                            nudge_y = 0.1, color = "Gray") + theme_minimal()
PCOA.Rel.plot.mucus
```

13.2 PCOA split by donor

```
# RelAbs <- decostand(data.frame(otu.table.t.ns.cpnr[1:142]), 2, method = "total")
RelAbs <- otu.table.t.ns.cpnr.relabun[1:142]

RelAbs.t <- as.data.frame(t(RelAbs))
  RelAbs.t.meta <- cbind(RelAbs.t, metadata[1:142,])

##split RelAbs by donor
RelAbs.t.meta.D1 <- dplyr::filter(RelAbs.t.meta,Donor ==1)
RelAbs.t.meta.D2 <- dplyr::filter(RelAbs.t.meta,Donor ==2)
```

```
#-------------------------------------------------------------------------------------------#
#create distance matrix of relative abundances of OTU-table
#computes dissimilarity indices
disjac.D1 <- vegan::vegdist(RelAbs.t.meta.D1[,1:28126], method="jaccard")

# disjac.cs <- vegdist(t(shared.cs), method="jaccard")
#computes dissimilarity indices
disjac.D2 <- vegan::vegdist(RelAbs.t.meta.D2[,1:28126], method="jaccard")

# disjac.cs <- vegdist(t(shared.cs), method="jaccard")


#-------------------------------------------------------------------------------------------#

# PCOA.Rel<- pcoa(disjac, correction="none", rn=NULL)
PCOA.Rel.D1<- prcomp(disjac.D1, center = TRUE, scale = TRUE)  #==> normalize or not????
PCOA.Rel.D1<- prcomp(disjac.D1)

metadata.D1 <- dplyr::filter(metadata[1:142,], Donor == 1)

PCOA.Rel.D2<- prcomp(disjac.D2, center = TRUE, scale = TRUE)  #==> normalize or not????
PCOA.Rel.D2<- prcomp(disjac.D2)

metadata.D2 <- dplyr::filter(metadata[1:142,], Donor == 2)

#-------------------------------------------------------------------------------------------#
# S3 method for pcoa
# biplot(PCOA.Rel.D1, Y=NULL, plot.axes = c(1,2), dir.axis1=1,
#        dir.axis2=1, rn=NULL, main=NULL)
# #plot using ggbiplot
# Rel.biplot.D1 <- ggbiplot(PCOA.Rel.D1, obs.scale = 1, var.scale = 1,
#                           groups = metadata$SHIME[1:142], ellipse = TRUE, circle = TRUE)

plot(PCOA.Rel.D1)
```

```
summary(PCOA.Rel.D1)

plot(PCOA.Rel.D2)
```

```
summary(PCOA.Rel.D2)

#-------------------------------------------------------------------------------------------#

#plot using ggplot
PCOA.Rel.data.D1 <- as.data.frame(PCOA.Rel.D1$rotation)[1:2]
PCOA.Rel.data.D1$Samplenr <- rownames(PCOA.Rel.data.D1)
PCOA.Rel.merge.D1 <- merge(PCOA.Rel.data.D1,metadata.D1,by = "Samplenr")

PCOA.Rel.plot1.D1 <- ggplot(PCOA.Rel.merge.D1,
                            aes(x = PC1, y = PC2, color = as.factor(Location))) +
  geom_point(size = 3)  + theme_bw() + labs(color = "Location",shape ="Donor")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
```

```
  scale_color_manual(values = c("#754ba6","#faed75","mediumblue", "orange"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
  #theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
PCOA.Rel.plot1.D1
```

```
PCOA.Rel.plot2.D1 <- ggplot(PCOA.Rel.merge.D1,
                            aes(x = PC1, y = PC2, color = as.factor(Treatment),
                                shape = as.factor(Location))) + geom_point(size = 3) +
  geom_text(aes(x = PC1,y= PC2, label = Timepoint), hjust = -1, vjust = -1)+ theme_bw() +
  labs(color = "Treatment",shape ="Donor")+ xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot2.D1
```

```
PCOA.Rel.plot3.D1 <- ggplot(PCOA.Rel.merge.D1, aes(x = PC1, y = PC2,
                                                   color = as.factor(Timepoint),
                                                   shape = as.factor(Donor))) +
  geom_point(size = 3)+ geom_text(aes(x = PC1,y= PC2, label = Timepoint),
                                  hjust = -1, vjust = -1) + theme_bw() +
  labs(color = "Timepoint",shape ="Donor")+ xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  scale_color_viridis(option="D",discrete = TRUE)+
  # scale_color_manual(values = c("#0c8560","#999999", "#d28eed", "#844da8" ,"#f2b385",
  #"#d97914", "#f5e2a2", "#e3d029", "#F781BF","#74dff7", "#125087"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
```

```
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot3.D1
```

```
PCOA.Rel.plot4.D1 <- ggplot(PCOA.Rel.merge.D1, aes(x = PC1, y = PC2,
                                                   color = as.factor(Treatment))) +
  geom_point(aes(size = Timepoint)) + geom_text(aes(x = PC1,y= PC2, label = Timepoint),
                                                hjust = -1, vjust = -1) +
  facet_grid(SHIME~Location)+ theme_bw() + labs(color = "Treatment")+
  xlab("PC1 (32.14%)") + ylab("PC2 (15.31%)")+
  # scale_color_viridis(option="D",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot4.D1
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA1_D1.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot1.D1
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA2_D1.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot2.D1
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA3_D1.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot3.D1
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA4_D1.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot4.D1
dev.off()

PCOA.Rel.merge.lumen <- dplyr::filter(PCOA.Rel.merge,Location == "lumen")
```

```
PCOA.Rel.plot.lumen <- ggplot(PCOA.Rel.merge.lumen, aes(x = PC1, y = PC2,
                                                        color = as.factor(Treatment),
                                                    shape = as.factor(SHIME), size = 4)) +
  geom_point()  + geom_text(aes(x = PC1, y = PC2,label= Timepoint),hjust = 0,
                            nudge_y = 0.1, color = "Gray") + theme_minimal()
PCOA.Rel.plot.lumen
```

```
PCOA.Rel.merge.mucus <- dplyr::filter(PCOA.Rel.merge,Location == "mucin")
PCOA.Rel.plot.mucus <- ggplot(PCOA.Rel.merge.mucus, aes(x = PC1, y = PC2,
                                                        color = as.factor(Treatment),
                                                    shape = as.factor(SHIME),size = 4)) +
  geom_point()  + geom_text(aes(x = PC1, y = PC2,label= Timepoint),hjust = 0,
                            nudge_y = 0.1, color = "Gray") + theme_minimal()
PCOA.Rel.plot.mucus
```

13.3 PCOA split by emulsifier

```
# RelAbs <- decostand(data.frame(otu.table.t.ns.cpnr[1:142]), 2, method = "total")
RelAbs <- otu.table.t.ns.cpnr.relabun[1:142]

RelAbs.t <- as.data.frame(t(RelAbs))
  RelAbs.t.meta <- cbind(RelAbs.t, metadata[1:142,])

##split RelAbs by donor
RelAbs.t.meta.RL.P80 <- dplyr::filter(RelAbs.t.meta,Emulsifier %in%
                                        c("Fecal","Control","RL","P80"))

#-------------------------------------------------------------------------------------------#
#create distance matrix of relative abundances of OTU-table
#computes dissimilarity indices
disjac.RL.P80 <- vegan::vegdist(RelAbs.t.meta.RL.P80[,1:28126], method="jaccard")
# disjac.cs <- vegdist(t(shared.cs), method="jaccard")

#-------------------------------------------------------------------------------------------#

# PCOA.Rel<- pcoa(disjac, correction="none", rn=NULL)

PCOA.Rel.RL.P80 <- prcomp(disjac.RL.P80, center = TRUE, scale = TRUE)
#==> normalize or not????
PCOA.Rel.RL.P80 <- prcomp(disjac.RL.P80)

metadata.RL.P80 <- dplyr::filter(metadata[1:142,], Emulsifier %in%
                                   c("Fecal","Control","RL","P80"))

#-------------------------------------------------------------------------------------------#
#PLOT RL P80
# S3 method for pcoa
# biplot(PCOA.Rel.RL.P80, Y=NULL, plot.axes = c(1,2), dir.axis1=1,
#        dir.axis2=1, rn=NULL, main=NULL)
# #plot using ggbiplot
# Rel.biplot.Soy <- ggbiplot(PCOA.Rel.RL.P80, obs.scale = 1, var.scale = 1,
#                            groups = metadata$SHIME[1:142], ellipse = TRUE, circle = TRUE)
```

```
plot(PCOA.Rel.RL.P80)
```

```
summary(PCOA.Rel.RL.P80)

#-------------------------------------------------------------------------------------------#

#plot using ggplot
PCOA.Rel.data.RL.P80 <- as.data.frame(PCOA.Rel.RL.P80$rotation)[1:2]
PCOA.Rel.data.RL.P80$Samplenr <- rownames(PCOA.Rel.data.RL.P80)
PCOA.Rel.merge.RL.P80 <- merge(PCOA.Rel.data.RL.P80,metadata.RL.P80,by = "Samplenr")

PCOA.Rel.plot1.RL.P80 <- ggplot(PCOA.Rel.merge.RL.P80, aes(x = PC1, y = PC2,
                                                           color = as.factor(Location),
                                                           shape = as.factor(Donor))) +
  geom_point(size = 3)  + theme_bw() + labs(color = "Location",shape ="Donor")+
  xlab("PC1 (22.66%)") + ylab("PC2 (13.84%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
  scale_color_manual(values = c("#754ba6","#faed75","mediumblue", "orange"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
  # theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
PCOA.Rel.plot1.RL.P80
```

```
PCOA.Rel.plot2.RL.P80 <- ggplot(PCOA.Rel.merge.RL.P80, aes(x = PC1, y = PC2,
                                                           color = as.factor(Treatment),
                                                           shape = as.factor(Donor))) +
  geom_point(size = 3) + geom_text(aes(x = PC1,y= PC2, label = Timepoint), hjust = -1,
                                   vjust = -1)+ theme_bw() + labs(color = "Treatment",
                                                                  shape ="Donor")+
  xlab("PC1 (22.66%)") + ylab("PC2 (13.84%)")+
  # scale_color_viridis(option="A",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8","#74dff7",
                                "#125087", "#f5e2a2", "#e3d029", "#d28eed", "#844da8",
                                "#F781BF","#f2b385", "#d97914"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
```

```
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot2.RL.P80
```

```
PCOA.Rel.plot3.RL.P80 <- ggplot(PCOA.Rel.merge.RL.P80, aes(x = PC1, y = PC2,
                                                           color = as.factor(Timepoint),
                                                           shape = as.factor(Donor))) +
  geom_point(size = 3)+ geom_text(aes(x = PC1,y= PC2, label = Timepoint), hjust = -1,
                                  vjust = -1) + theme_bw() + labs(color = "Timepoint",
                                                                  shape ="Donor")+
  xlab("PC1 (22.66%)") + ylab("PC2 (13.84%)")+
  scale_color_viridis(option="D",discrete = TRUE)+
  # scale_color_manual(values = c("#0c8560","#999999", "#d28eed", "#844da8" ,"#f2b385",
  ##d97914", "#f5e2a2", "#e3d029", "#F781BF","#74dff7", "#125087"))+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot3.RL.P80
```

```
PCOA.Rel.plot4.RL.P80 <- ggplot(PCOA.Rel.merge.RL.P80, aes(x = PC1, y = PC2,
                                                           color = as.factor(Treatment))) +
  geom_point(aes(size = Timepoint)) + geom_text(aes(x = PC1,y= PC2, label = Timepoint),
                                                hjust = -1, vjust = -1) +
  facet_grid(Donor~Location)+ theme_bw() + labs(color = "Treatment")+ xlab("PC1 (22.66%)") +
  ylab("PC2 (13.84%)")+
  # scale_color_viridis(option="D",discrete = TRUE)+
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))+
  xlim(-.15,0.18)+
  ylim(-0.2,0.25)+
  theme(axis.text.x = element_text(size = 10,family='sans',angle=0),
        axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =14,family='sans',angle=0),
        axis.title.y = element_text(size=14,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0),
        # legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=12,angle=0,family='sans'),
        # legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())
PCOA.Rel.plot4.RL.P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA1_RL.P80.tiff",
```

```
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot1.RL.P80
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA2_RL.P80.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot2.RL.P80
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA3_RL.P80.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot3.RL.P80
dev.off()

dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_PCoA4_RL.P80.tiff",
     width=8*dpi,height=6*dpi,res=dpi,compression="lzw")
PCOA.Rel.plot4.RL.P80
dev.off()
```

#RDA

13.3.1 OTU - global

```
## partial distance based RDA
Xvars.rda <- metadata[1:142,]
Yvars.rda <- as.matrix(t(RelAbs))

#-------------------------------------------------------------------------------------------#
rda_OTU_Rel1 <- capscale(Yvars.rda ~ Donor + Treatment + Timepoint + Location,
                         data = Xvars.rda, distance = "jaccard")
anova(rda_OTU_Rel1, by = "term", permutations = 199)
 #           Df SumOfSqs       F Pr(>F)
#Donor        1   6.3905 25.6507  0.005 **
#Treatment    7   4.1266  2.3663  0.005 **
#Timepoint    8   7.8554  3.9414  0.005 **
#Location     1   1.9548  7.8465  0.005 **
#Residual   124  30.8927

# plot(rda_OTU_Rel1)
# points(rda_OTU_Rel1, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_OTU_Rel1, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19, bty = "n")

#-------------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_OTU_Rel1 <- summary(rda_OTU_Rel1)
  #CAP1 = 13,28%
  #CAP2 = 6,76,%

species1 <- data.frame(smry_rda_OTU_Rel1$species[,1:2])  #we only need the first 2
```

```
                                                #coordinates for both species and sites.
species1$Genus <- taxonomy.np.ns$Genus

sites1 <- data.frame(smry_rda_OTU_Rel1$sites[,1:2])
sites.meta1 <- cbind(sites1, metadata[1:142,])
centroids1 <- data.frame(smry_rda_OTU_Rel1$centroids[,1:2])
rownames(centroids1) <- c("D1","D2","Control - 0","Fecal - 0","P80 - 0.05","P80 - 0.5",
                          "RL - 0.05","RL - 0.5","SoyLec - 0.05","SoyLec - 0.5","T2","T4","T9","T10","T
# biplot <- data.frame(smry_rda_OTU_Rel1$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
sites.meta1$Emulsifier <- as.factor(sites.meta1$Emulsifier)
sites.meta1$SHIME <- as.factor(sites.meta1$SHIME)

# Colored by treatment

rda_Caps_plot_Treatment <- ggplot(sites.meta1, aes(x=CAP1, y=CAP2, colour = Treatment)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                               "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                               "#d97914"))
rda_Caps_plot_Treatment
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Treatment  <- rda_Caps_plot_Treatment +
  geom_segment(data=species1[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Treatment
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Treatment  <- rda_Caps_biplot_Treatment +
  geom_segment(data=centroids1[3:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1[3:10,],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1[3:10,]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,76%)") +
  xlab("CAP1 (13,28%)")+
  theme_bw()
rda_Caps_triplot1_Treatment
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_Rel_RDA_Glo
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Treatment
```

```
dev.off()


# Colored by Donor

rda_Caps_plot_Donor <- ggplot(sites.meta1, aes(x=CAP1, y=CAP2, colour = Donor)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                               "#844da8","#74dff7","Black","#999999",  "#e3d029",
                               "#d97914"))
rda_Caps_plot_Donor
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Donor  <- rda_Caps_plot_Donor +
  geom_segment(data=species1[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Donor
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Donor  <- rda_Caps_biplot_Donor +
  geom_segment(data=centroids1[c(1,2),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1[c(1,2),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1[c(1,2),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,76%)") +
  xlab("CAP1 (13,28%)")+
  theme_bw()
rda_Caps_triplot1_Donor
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_Rel_RDA_Glo
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Donor
dev.off()


# Colored by Gut Location

rda_Caps_plot_Location <- ggplot(sites.meta1, aes(x=CAP1, y=CAP2, colour = Location)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
```

```
  scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                                "#844da8","#74dff7","Black","#999999","#e3d029",
                                "#d97914"))
rda_Caps_plot_Location
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Location  <- rda_Caps_plot_Location +
  geom_segment(data=species1[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Location
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Location  <- rda_Caps_biplot_Location +
  geom_segment(data=centroids1[c(20,21),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1[c(20,21),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1[c(20,21),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,76%)") +
  xlab("CAP1 (13,28%)")+
  theme_bw()
rda_Caps_triplot1_Location
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_Rel_RDA_Glo
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Location
dev.off()
```

13.3.2 Genus - global

```
## partial distance based RDA
Xvars.rda <- metadata[1:142,]
Yvars.rda.G <- as.matrix(t(otu.table.t.ns.cpnr.relabun.Gr[2:143]))

#-------------------------------------------------------------------------------------#
rda_OTU_Rel1_G <- capscale(Yvars.rda.G ~ Donor + Treatment + Timepoint + Location,
                           data = Xvars.rda, distance = "jaccard")
anova(rda_OTU_Rel1_G, by = "term", permutations = 199)
#            Df SumOfSqs       F Pr(>F)
# Donor       1   5.1757 31.4876  0.005 **
# Treatment   7   2.6852  2.3337  0.005 **
# Timepoint   8   6.9919  5.3171  0.005 **
# Location    1   1.8947 11.5271  0.005 **
# Residual  124  20.3822
# ---
```

```
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

plot(rda_OTU_Rel1_G)
```

```
# points(rda_OTU_Rel1_G, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_OTU_Rel1_G, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19, bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_OTU_Rel1_G <- summary(rda_OTU_Rel1_G)
  #CAP1 = 17,18%
  #CAP2 = 9,59%

species1G <- data.frame(smry_rda_OTU_Rel1_G$species[,1:2])  #we only need the first 2
                                                            #coordinates for both species and sites.
species1G$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites1G <- data.frame(smry_rda_OTU_Rel1_G$sites[,1:2])
sites.meta1G <- cbind(sites1G, metadata[1:142,])
centroids1G <- data.frame(smry_rda_OTU_Rel1_G$centroids[,1:2])
rownames(centroids1G) <- c("D1","D2","Control - 0","Fecal - 0","P80 - 0.05","P80 - 0.5",
                           "RL - 0.05","RL - 0.5","SoyLec - 0.05","SoyLec - 0.5","T2","T4","T9","T10","T
# biplot <- data.frame(smry_rda_OTU_Rel1$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
sites.meta1G$Emulsifier <- as.factor(sites.meta1$Emulsifier)
sites.meta1G$SHIME <- as.factor(sites.meta1$SHIME)

# Colored by treatment

rda_Caps_plot_Treatment_G <- ggplot(sites.meta1G, aes(x=CAP1, y=CAP2,colour=Treatment)) +
  geom_point(size = 6) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                               "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                               "#d97914"))
rda_Caps_plot_Treatment_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Treatment_G  <- rda_Caps_plot_Treatment_G +
  geom_segment(data=species1G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=5)+
  theme_bw()
rda_Caps_biplot_Treatment_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Treatment_G  <- rda_Caps_biplot_Treatment_G +
```

```
  geom_segment(data=centroids1G[3:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G[3:10,],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G[3:10,]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=5)+
  ylab("CAP2 (9,59%)") +
  xlab("CAP1 (17,18%)")+
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))+
  xlim(-3,3)
rda_Caps_triplot1_Treatment_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Treatmen
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Treatment_G
dev.off()


# Colored by Donor

rda_Caps_plot_Donor_G <- ggplot(sites.meta1G, aes(x=CAP1, y=CAP2, colour = Donor)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                               "#844da8","#74dff7","Black","#999999",  "#e3d029",
                               "#d97914"))
rda_Caps_plot_Donor_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Donor_G  <- rda_Caps_plot_Donor_G +
  geom_segment(data=species1G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Donor_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Donor_G  <- rda_Caps_biplot_Donor_G +
  geom_segment(data=centroids1G[c(1,2),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G[c(1,2),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G[c(1,2),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
```

```
  ylab("CAP2 (9,59%)") +
  xlab("CAP1 (17,18%)")+
  theme_bw()
rda_Caps_triplot1_Donor_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Donor_G
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Donor_G
dev.off()

# Colored by Gut Location

rda_Caps_plot_Location_G <- ggplot(sites.meta1G, aes(x=CAP1, y=CAP2, colour = Location))+
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                                "#844da8","#74dff7","Black","#999999",  "#e3d029",
                                "#d97914"))
rda_Caps_plot_Location_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Location_G  <- rda_Caps_plot_Location +
  geom_segment(data=species1G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Location_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Location_G  <- rda_Caps_biplot_Location_G +
  geom_segment(data=centroids1G[c(20,21),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G[c(20,21),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G[c(20,21),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (9,59%)") +
  xlab("CAP1 (17,18%)")+
  theme_bw()
rda_Caps_triplot1_Location_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Locatio
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Location_G
dev.off()
```

13.4 Genus - Treatment

```
## RDA based with all but  Treatment conditioned out
rda_OTU_Rel3_G <- capscale(Yvars.rda.G ~ Treatment + Condition(Donor) +
                             Condition(Timepoint) + Condition(Location),
                           data = Xvars.rda, distance = "jaccard")
anova(rda_OTU_Rel3_G, by = "term", permutations = 199)
#            Df SumOfSqs      F Pr(>F)
# Treatment   6   2.4402 2.4742  0.005 **
# Residual  124  20.3822
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# plot(rda_OTU_Rel3_G)
# points(rda_OTU_Rel3_G, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_OTU_Rel3_G, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_OTU_Rel3_G <- summary(rda_OTU_Rel3_G)
  #CAP1=6,04%
  #CAP2=1,96%

Runadj_OTU_Rel3 <- vegan::RsquareAdj(rda_OTU_Rel3_G)$r.squared # 0.068
Radj_OTU_Rel3 <- RsquareAdj(rda_OTU_Rel3_G)$adj.r.squared # 0.046

species3G <- data.frame(smry_rda_OTU_Rel3_G$species[,1:2])  #we only need the first 2
                                                            #coordinates for both species and sites.
species3G$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites3G <- data.frame(smry_rda_OTU_Rel3_G$sites[,1:2])
sites.meta3G <- cbind(sites3G, metadata[1:142,])
centroids3G <- data.frame(smry_rda_OTU_Rel3_G$centroids[,1:2])
rownames(centroids3G) <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05","RL - 0.5",
                           "SoyLec - 0.05","SoyLec - 0.5")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta3G$Donor <- as.factor(sites.meta3G$Donor)

rda_Caps_plot_3_G <- ggplot(sites.meta3G, aes(x=CAP1, y=CAP2, colour = Treatment)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))
```

```
rda_Caps_plot_3_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_3_G <- rda_Caps_plot_3_G +
  geom_segment(data=species3G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species3G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_3_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_3_G <- rda_Caps_biplot_3_G +
  geom_segment(data=centroids3G, aes(x=0, xend=CAP1, y=0, yend=CAP2),
                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids3G,
            aes(x=CAP1,y=CAP2, label=rownames(centroids3G),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
   ylab("CAP2 (1,96%)") +
   xlab("CAP1 (6,04%)")+
   theme_bw()
rda_Caps_triplot_3_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Treatmer
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_3_G
dev.off()
```

#RDA_RL_P80 ## All factors

```
## partial distance based RDA
Xvars.rda.RL.P80 <- RelAbs.t.meta.RL.P80[,28127:28136]
Yvars.rda.RL.P80 <- as.matrix(RelAbs.t.meta.RL.P80[,1:28126])

#-------------------------------------------------------------------------------------#
rda_OTU_Rel1_RL.P80 <- capscale(Yvars.rda.RL.P80 ~ Donor + Treatment + Timepoint +
                                  Location, data = Xvars.rda.RL.P80, distance = "jaccard")
anova(rda_OTU_Rel1_RL.P80, by = "term", permutations = 199)
#           Df SumOfSqs       F Pr(>F)
# Donor      1   4.0503 15.5556  0.005 **
# Treatment  5   3.2576  2.5022  0.005 **
# Timepoint  8   5.9545  2.8586  0.005 **
# Location   1   1.3560  5.2079  0.005 **
# Residual  86  22.3921
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

# plot(rda_OTU_Rel1_RL.P80)
# points(rda_OTU_Rel1_RL.P80, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_OTU_Rel1_RL.P80, display = "species", cex = 0.8, col = "red")
```

```
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19, bty = "n")

#-------------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_OTU_Rel1_RL.P80 <- summary(rda_OTU_Rel1_RL.P80)
  #CAP1 = 11,98%
  #CAP2 = 6,66%

species1_RL.P80 <- data.frame(smry_rda_OTU_Rel1_RL.P80$species[,1:2]) #we only need
                                # the first 2 coordinates for both species and sites.

species1_RL.P80$Genus <- taxonomy.np.ns$Genus

sites1_RL.P80  <- data.frame(smry_rda_OTU_Rel1_RL.P80$sites[,1:2])
sites.meta1_RL.P80 <- cbind(sites1_RL.P80,RelAbs.t.meta.RL.P80[,28127:28136])
centroids1_RL.P80  <- data.frame(smry_rda_OTU_Rel1_RL.P80$centroids[,1:2])
rownames(centroids1_RL.P80 ) <- c("D1","D2","Control - 0","Fecal - 0","P80 - 0.05",
                                  "P80 - 0.5","RL - 0.05","RL - 0.5","T2","T4","T9",
                                  "T10","T11","T12","T14","T16","T18","Lumen","Mucus")
# biplot <- data.frame(smry_rda_OTU_Rel1$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
sites.meta1_RL.P80$Emulsifier <- as.factor(sites.meta1_RL.P80$Emulsifier)
sites.meta1_RL.P80$SHIME <- as.factor(sites.meta1_RL.P80$SHIME)

# Colored by treatment
rda_Caps_plot_Treatment_RL.P80  <- ggplot(sites.meta1_RL.P80, aes(x=CAP1, y=CAP2,
                                                                  colour = Treatment)) +
  geom_point(size = 6) +
  geom_text(aes(x= CAP1, y = CAP2, label=Timepoint,
                # hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)
                                                  )) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))
rda_Caps_plot_Treatment_RL.P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Treatment_RL.P80   <- rda_Caps_plot_Treatment_RL.P80   +
  geom_segment(data=species1_RL.P80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1_RL.P80[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1_RL.P80)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Treatment_RL.P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Treatment_RL.P80 <- rda_Caps_biplot_Treatment_RL.P80   +
  geom_segment(data=centroids1_RL.P80[3:6,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
```

```
  geom_text(data=centroids1_RL.P80[3:6,],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1_RL.P80[3:6,]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,66%)") +
  xlab("CAP1 (11,98%)")+
  xlim(-2,1.5)+
  ylim(-2.1,2)+
  theme_bw() + theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot1_Treatment_RL.P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Gen
     width = 3000,height = 2500,
     res =300)
rda_Caps_triplot1_Treatment_RL.P80
dev.off()

# Colored by Donor
rda_Caps_plot_Donor_RL.P80 <- ggplot(sites.meta1_RL.P80, aes(x=CAP1, y=CAP2,
                                                             colour = Donor)) +
  geom_point(size = 6) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                               "#844da8","#74dff7","Black","#999999","#e3d029",
                               "#d97914"))
rda_Caps_plot_Donor_RL.P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Donor_RL.P80  <- rda_Caps_plot_Donor_RL.P80 +
  geom_segment(data=species1_RL.P80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1_RL.P80[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1_RL.P80)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Donor_RL.P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Donor_RL.P80  <- rda_Caps_biplot_Donor_RL.P80 +
  geom_segment(data=centroids1_RL.P80[c(1,2),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1_RL.P80[c(1,2),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1_RL.P80[c(1,2),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,66%)")+
```

```
  xlab("CAP1 (11,98%)")+
  xlim(-2,1.5)+
  ylim(-2.1,2)+
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))
rda_Caps_triplot1_Donor_RL.P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Gen
     width = 3000,height = 2500,
     res =300)
rda_Caps_triplot1_Donor_RL.P80
dev.off()

# Colored by Gut Location
rda_Caps_plot_Location_RL.P80<- ggplot(sites.meta1_RL.P80,
                                       aes(x=CAP1, y=CAP2, colour = Location)) +
  geom_point(size = 6) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                                "#844da8","#74dff7","Black","#999999","#e3d029",
                                "#d97914"))
rda_Caps_plot_Location_RL.P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Location_RL.P80  <- rda_Caps_plot_Location_RL.P80 +
  geom_segment(data=species1_RL.P80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1_RL.P80[1:10,],
            aes(x=CAP1,y=CAP2, label=rownames(species1_RL.P80)[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Location_RL.P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Location_RL.P80 <- rda_Caps_biplot_Location_RL.P80 +
  geom_segment(data=centroids1_RL.P80[c(18,19),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1_RL.P80[c(18,19),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1_RL.P80[c(18,19),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (6,66%)")+
  xlab("CAP1 (11,98%)")+
  xlim(-2,1.5)+
  ylim(-2.1,2)+
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
```

```
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))
rda_Caps_triplot1_Location_RL.P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Gen
     width = 3000,height = 2500,
     res =300)
rda_Caps_triplot1_Location_RL.P80
dev.off()
```

13.5 Genus - global

```
RelAbs.RL.P80 <- otu.table.t.ns.cpnr.relabun[1:142]
RelAbs.t <- as.data.frame(t(RelAbs))

RelAbs.Gr.meta <- cbind(t(otu.table.t.ns.cpnr.relabun.Gr[2:143]), metadata[1:142,])

##split RelAbs by donor
RelAbs.Gr.meta.RL.P80 <- dplyr::filter(RelAbs.Gr.meta,Emulsifier %in%
                                       c("Fecal","Control","P80","RL"))

## partial distance based RDA
Xvars.rda.G.RL.P80 <- RelAbs.Gr.meta.RL.P80[199:209]
Yvars.rda.G.RL.P80 <- as.matrix(RelAbs.Gr.meta.RL.P80[1:199])

#-------------------------------------------------------------------------------------------#
rda_G_Rel1_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Donor + Treatment + Timepoint +
                                Location, data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Rel1_RL_P80, by = "term", permutations = 199)
#            Df SumOfSqs       F Pr(>F)
# Donor       1   3.1117 18.3727  0.005 **
# Treatment   5   2.1662  2.5581  0.005 **
# Timepoint   8   5.3500  3.9486  0.005 **
# Location    1   1.4303  8.4450  0.005 **
# Residual   86  14.5652
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

  #P_Donor = 0,005 **
  #P_Treatment = 0,005 **
  #P_Timepoint = 0,005 **
  #P_Location = 0,005 **

# plot(rda_G_Rel1_RL_P80)
# points(rda_G_Rel1_RL_P80, pch = 19, col = colvec[Xvars.rda.G.RL.P80$Donor])
# text(rda_G_Rel1_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda.G.RL.P80$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
```

```
smry_rda_G_Rel1_RL_P80 <- summary(rda_G_Rel1_RL_P80)
  #CAP1 = 15,75%
  #CAP2 = 10,09%

species1G_RL_P80  <- data.frame(smry_rda_G_Rel1_RL_P80 $species[,1:2])  #we only need
                                #the first 2 coordinates for both species and sites.
species1G_RL_P80 $Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites1G_RL_P80  <- data.frame(smry_rda_G_Rel1_RL_P80$sites[,1:2])
sites.meta1G_RL_P80  <- cbind(sites1G_RL_P80, RelAbs.Gr.meta.RL.P80[199:209])
centroids1G_RL_P80  <- data.frame(smry_rda_G_Rel1_RL_P80$centroids[,1:2])
rownames(centroids1G_RL_P80 ) <- c("D1","D2","Control - 0","Fecal - 0","P80 - 0.05",
                                "P80 - 0.5","RL - 0.05","RL - 0.5","T2","T4","T9",
                                "T10","T11","T12","T14","T16","T18","Lumen","Mucus")
# biplot <- data.frame(smry_rda_OTU_Rel1$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
sites.meta1G_RL_P80$Emulsifier <- as.factor(sites.meta1G_RL_P80$Emulsifier)
sites.meta1G_RL_P80$SHIME <- as.factor(sites.meta1G_RL_P80$SHIME)

# Colored by treatment
rda_Caps_plot_Treatment_G_RL_P80 <- ggplot(sites.meta1G_RL_P80, aes(x=CAP1, y=CAP2,
                                                                    colour = Treatment)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))
rda_Caps_plot_Treatment_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Treatment_G_RL_P80 <- rda_Caps_plot_Treatment_G_RL_P80 +
  geom_segment(data=species1G_RL_P80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G_RL_P80[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Treatment_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Treatment_G_RL_P80  <- rda_Caps_biplot_Treatment_G_RL_P80 +
  geom_segment(data=centroids1G_RL_P80[3:8,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G_RL_P80[3:8,],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G_RL_P80[3:8,]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (10,09%)") +
  xlab("CAP1 (15,75%)")+
  theme_bw()
```

```
rda_Caps_triplot1_Treatment_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Gen
     width = 3000,height = 2500,
     res =300)
rda_Caps_triplot1_Treatment_G_RL_P80
dev.off()

# Colored by Donor
rda_Caps_plot_Donor_G <- ggplot(sites.meta1G_RL_P80, aes(x=CAP1, y=CAP2, colour = Donor)) +
  geom_point(size = 6) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                               "#844da8","#74dff7","Black","#999999",  "#e3d029",
                               "#d97914"))
rda_Caps_plot_Donor_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Donor_G  <- rda_Caps_plot_Donor_G +
  geom_segment(data=species1G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Donor_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Donor_G  <- rda_Caps_biplot_Donor_G +
  geom_segment(data=centroids1G[c(1,2),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G[c(1,2),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G[c(1,2),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (9,59%)") +
  xlab("CAP1 (17,18%)")+
  theme_bw()
rda_Caps_triplot1_Donor_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Donor_G
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Donor_G
dev.off()

# Colored by Gut Location
rda_Caps_plot_Location_G <- ggplot(sites.meta1G_RL_P80, aes(x=CAP1, y=CAP2,
                                                           colour = Location)) +
  geom_point(size = 6) +
```

```
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                                "#844da8","#74dff7","Black","#999999",  "#e3d029",
                                "#d97914"))
rda_Caps_plot_Location_G
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Location_G  <- rda_Caps_plot_Location_G +
  geom_segment(data=species1G[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Location_G
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Location_G  <- rda_Caps_biplot_Location_G +
  geom_segment(data=centroids1G[c(20,21),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G[c(20,21),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G[c(20,21),]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (9,59%)") +
  xlab("CAP1 (17,18%)")+
  theme_bw()
rda_Caps_triplot1_Location_G
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Locatio
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Location_G
dev.off()
```

13.6 Genus - Treatment

```
## RDA based with all but  Donor conditioned out
rda_G_Rel3_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Treatment + Condition(Donor) +
                                Condition(Timepoint) + Condition(Location),
                              data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Rel3_RL_P80, by = "term", permutations = 199)
#           Df SumOfSqs      F Pr(>F)
# Treatment  4    1.920 2.8342  0.005 **
# Residual  86   14.565
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#significance Treatment = 0.005 *
```

```
# plot(rda_G_Rel3_RL_P80)
# points(rda_G_Rel3_RL_P80, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_G_Rel3_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19, bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Rel3_RL_P80 <- summary(rda_G_Rel3_RL_P80)
  #CAP1=8,23%
  #CAP2=1,59%

Runadj_G_Rel3_RL_P80 <- vegan::RsquareAdj(rda_G_Rel3_RL_P80)$r.squared #=> 7,39%
Radj_G_Rel3_RL_P80 <- RsquareAdj(rda_G_Rel3_RL_P80)$adj.r.squared #=> 5,49%

species3G_RL_P80 <- data.frame(smry_rda_G_Rel3_RL_P80$species[,1:2])  #we only need the
                                        #first 2 coordinates for both species and sites.
species3G_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites3G_RL_P80 <- data.frame(smry_rda_G_Rel3_RL_P80$sites[,1:2])
sites.meta3G_RL_P80 <- cbind(sites3G_RL_P80, RelAbs.Gr.meta.RL.P80[199:209])
centroids3G_RL_P80 <- data.frame(smry_rda_G_Rel3_RL_P80$centroids[,1:2])
rownames(centroids3G_RL_P80) <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05",
                                  "RL - 0.5")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
# "Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta3G_RL_P80$Donor <- as.factor(sites.meta3G_RL_P80$Donor)

rda_Caps_plot_3_G_RL_P80 <- ggplot(sites.meta3G_RL_P80, aes(x=CAP1, y=CAP2,
                                                            colour = Treatment)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("Black","#999999", "#d28eed", "#844da8" ,"#74dff7",
                                "#125087","#f5e2a2", "#e3d029", "#F781BF","#f2b385",
                                "#d97914"))
rda_Caps_plot_3_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_3_G_RL_P80 <- rda_Caps_plot_3_G_RL_P80 +
  # geom_text(data = sites.meta3G_RL_P80, aes(x= CAP1, y = CAP2,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)))) +
  geom_segment(data=species3G_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species3G_RL_P80[1:5,],
            aes(x=CAP1,y=CAP2, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
```

```
  theme_bw()
rda_Caps_biplot_3_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_3_G_RL_P80 <- rda_Caps_biplot_3_G_RL_P80 +
  # geom_segment(data=centroids3G_RL_P80, aes(x=0, xend=CAP1, y=0, yend=CAP2),
  #               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids3G_RL_P80,
            aes(x=CAP1,y=CAP2, label=rownames(centroids3G_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
   ylab("CAP2 (1,59%)") +
   xlab("CAP1 (8,23%)")+
   xlim(-3,4)+
   ylim(-3,4)+
   theme_bw()+ theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_3_G_RL_P80
```

```
rda_Caps_triplot_3_G_RL_P80 <- rda_Caps_triplot_3_G_RL_P80 +
  annotate(geom="text", x=2.2, y=3.5, label="R²_adj = 5,49",
              color="black", size = 5)
rda_Caps_triplot_3_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Trea
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_3_G_RL_P80
dev.off()
```

13.7 Genus - Donor

```
## RDA based with all but  Donor conditioned out
rda_G_Rel2_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Donor + Condition(Treatment) +
                                Condition(Timepoint) + Condition(Location),
                              data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Rel2_RL_P80, by = "term", permutations = 199)
#          Df SumOfSqs      F Pr(>F)
# Donor     1   1.7682 10.441  0.005 **
# Residual 86  14.5652
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#significance Treatment = 0.005 *

# plot(rda_G_Rel2_RL_P80)
# points(rda_G_Rel2_RL_P80, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_G_Rel2_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19,
#        bty = "n")
```

```
#-------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Rel2_RL_P80 <- summary(rda_G_Rel2_RL_P80)
  #CAP1=10,83%
  #CAP2=17.27%

Runadj_G_Rel2_RL_P80 <- vegan::RsquareAdj(rda_G_Rel2_RL_P80)$r.squared #=> 6,80%
Radj_G_Rel2_RL_P80 <- RsquareAdj(rda_G_Rel2_RL_P80)$adj.r.squared #=> 7.18%

species2G_RL_P80 <- data.frame(smry_rda_G_Rel2_RL_P80$species[,1:2])  #we only need the
                                          #first 2 coordinates for both species and sites.
species2G_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites2G_RL_P80 <- data.frame(smry_rda_G_Rel2_RL_P80$sites[,1:2])
sites.meta2G_RL_P80 <- cbind(sites2G_RL_P80, RelAbs.Gr.meta.RL.P80[199:209])
centroids2G_RL_P80 <- data.frame(smry_rda_G_Rel2_RL_P80$centroids[,1:2])
rownames(centroids2G_RL_P80) <- c("Donor 1","Donor 2")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta2G_RL_P80$Donor <- as.factor(sites.meta2G_RL_P80$Donor)

rda_Caps_plot_2_G_RL_P80 <- ggplot(sites.meta2G_RL_P80, aes(x=CAP1, y=MDS1,
                                                            colour = Donor)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999", "#d28eed", "#844da8" ,"#74dff7", "#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385", "#d97914"))
rda_Caps_plot_2_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_2_G_RL_P80 <- rda_Caps_plot_2_G_RL_P80 +
  # geom_text(data = sites.meta2G_RL_P80, aes(x= CAP1, y = MDS1,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(MDS1)))) +
  geom_segment(data=species2G_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=MDS1),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species2G_RL_P80[1:5,],
            aes(x=CAP1,y=MDS1, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_2_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_2_G_RL_P80 <- rda_Caps_biplot_2_G_RL_P80 +
  # geom_segment(data=centroids2G_RL_P80, aes(x=0, xend=CAP1, y=0, yend=MDS1),
  #                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids2G_RL_P80,
```

```
            aes(x=CAP1,y=MDS1, label=rownames(centroids2G_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="blue", size=4)+
   ylab("MDS1 (17.27%)") +
   xlab("CAP1 (10,83%)")+
   xlim(-2.5,2.5)+
   ylim(-2,2.5)+
   theme_bw()+ theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_2_G_RL_P80
```

```
rda_Caps_triplot_2_G_RL_P80 <- rda_Caps_triplot_2_G_RL_P80 +
  annotate(geom="text", x=1.5, y=2.4, label="R²_adj = 7,18",
              color="black", size = 5)
rda_Caps_triplot_2_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Don
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_2_G_RL_P80
dev.off()
```

13.8 Genus - Location

```
## RDA based with all but  Donor conditioned out
rda_G_Rel4_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Location + Condition(Donor) +
                                Condition(Timepoint) + Condition(Treatment),
                              data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Rel4_RL_P80, by = "term", permutations = 199)
#          Df SumOfSqs     F Pr(>F)
# Location  1   1.4303 8.445  0.005 **
# Residual 86  14.5652
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#significance Treatment = 0.005 *

# plot(rda_G_Rel4_RL_P80)
# points(rda_G_Rel4_RL_P80, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_G_Rel4_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Rel4_RL_P80 <- summary(rda_G_Rel4_RL_P80)
  #CAP1=8,94%
  #CAP2=17,6%

Runadj_G_Rel4_RL_P80 <- vegan::RsquareAdj(rda_G_Rel4_RL_P80)$r.squared #=> 5.50%
Radj_G_Rel4_RL_P80 <- RsquareAdj(rda_G_Rel4_RL_P80)$adj.r.squared #=> 5,67%
```

```
species4G_RL_P80 <- data.frame(smry_rda_G_Rel4_RL_P80$species[,1:2])  #we only need the
                                        #first 2 coordinates for both species and sites.
species4G_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites4G_RL_P80 <- data.frame(smry_rda_G_Rel4_RL_P80$sites[,1:2])
sites.meta4G_RL_P80 <- cbind(sites4G_RL_P80, RelAbs.Gr.meta.RL.P80[199:209])
centroids4G_RL_P80 <- data.frame(smry_rda_G_Rel4_RL_P80$centroids[,1:2])
rownames(centroids4G_RL_P80) <- c("lumen","mucus")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta4G_RL_P80$Donor <- as.factor(sites.meta4G_RL_P80$Donor)

rda_Caps_plot_4_G_RL_P80 <- ggplot(sites.meta4G_RL_P80,
                                   aes(x=CAP1, y=MDS1, colour = Location)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999", "#d28eed", "#844da8" ,"#74dff7", "#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385", "#d97914"))
rda_Caps_plot_4_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_4_G_RL_P80 <- rda_Caps_plot_4_G_RL_P80 +
  # geom_text(data = sites.meta4G_RL_P80, aes(x= CAP1, y = MDS1,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(MDS1)))) +
  geom_segment(data=species4G_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=MDS1),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species4G_RL_P80[1:5,],
            aes(x=CAP1,y=MDS1, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_4_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_4_G_RL_P80 <- rda_Caps_biplot_4_G_RL_P80 +
  # geom_segment(data=centroids4G_RL_P80, aes(x=0, xend=CAP1, y=0, yend=MDS1),
  #                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids4G_RL_P80,
            aes(x=CAP1,y=MDS1, label=rownames(centroids4G_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="blue", size=4)+
  ylab("MDS1 (17,60%)") +
  xlab("CAP1 (8,94%)")+
  xlim(-3,3)+
  ylim(-2.5,2.5)+
  labs(colour = "Environment")+
```

```
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))
rda_Caps_triplot_4_G_RL_P80
```

```
rda_Caps_triplot_4_G_RL_P80 <- rda_Caps_triplot_4_G_RL_P80 +
  annotate(geom="text", x=1.5, y=2, label="R²_adj = 5,67",
              color="black", size = 5)
rda_Caps_triplot_4_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Loca
     width = 3000,height = 2500,
     res =300)
rda_Caps_triplot_4_G_RL_P80
dev.off()
```

13.9 Genus - Timepoint

```
## RDA based with all but  Donor conditioned out
rda_G_Rel5_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~  Timepoint + Condition(Donor) +
                                Condition(Treatment) + Condition(Location),
                              data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Rel5_RL_P80, by = "term", permutations = 199)
#           Df SumOfSqs      F Pr(>F)
# Timepoint  8   4.4243 3.2654  0.005 **
# Residual  86  14.5652
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#significance Treatment = 0.005 *

# plot(rda_G_Rel5_RL_P80)
# points(rda_G_Rel5_RL_P80, pch = 19, col = colvec[Xvars.rda$Donor])
# text(rda_G_Rel5_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Rel5_RL_P80 <- summary(rda_G_Rel5_RL_P80)
  #CAP1=11.37%
  #CAP2=5.43%

Runadj_G_Rel5_RL_P80 <- vegan::RsquareAdj(rda_G_Rel5_RL_P80)$r.squared #=> 17,03%
Radj_G_Rel5_RL_P80 <- RsquareAdj(rda_G_Rel5_RL_P80)$adj.r.squared #=> 12,94%

species5G_RL_P80 <- data.frame(smry_rda_G_Rel5_RL_P80$species[,1:2])  #we only need the
                                      #first 2 coordinates for both species and sites.
species5G_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites5G_RL_P80 <- data.frame(smry_rda_G_Rel5_RL_P80$sites[,1:2])
```

```
sites.meta5G_RL_P80 <- cbind(sites5G_RL_P80, RelAbs.Gr.meta.RL.P80[199:209])
centroids5G_RL_P80 <- data.frame(smry_rda_G_Rel5_RL_P80$centroids[,1:2])
rownames(centroids5G_RL_P80) <- c("T2","T4","T9","T10","T11","T12","T14","T16","T18")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta5G_RL_P80$Donor <- as.factor(sites.meta5G_RL_P80$Donor)

rda_Caps_plot_5_G_RL_P80 <- ggplot(sites.meta5G_RL_P80, aes(x=CAP1, y=CAP2,
                                                            colour = Timepoint)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999", "#d28eed", "#844da8" ,"#74dff7","#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385", "#d97914"))
rda_Caps_plot_5_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_5_G_RL_P80 <- rda_Caps_plot_5_G_RL_P80 +
  # geom_text(data = sites.meta5G_RL_P80, aes(x= CAP1, y = CAP2,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)))) +
  geom_segment(data=species5G_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species5G_RL_P80[1:5,],
            aes(x=CAP1,y=CAP2, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_5_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_5_G_RL_P80 <- rda_Caps_biplot_5_G_RL_P80 +
  # geom_segment(data=centroids5G_RL_P80, aes(x=0, xend=CAP1, y=0, yend=CAP2),
  #               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids5G_RL_P80,
            aes(x=CAP1,y=CAP2, label=rownames(centroids5G_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
   ylab("CAP2 (5,43%)") +
   xlab("CAP1 (11,37%)")+
  xlim(-5,4)+
  ylim(-3,2)+
   theme_bw()+ theme(axis.title.x=element_text(size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_5_G_RL_P80
```

```
rda_Caps_triplot_5_G_RL_P80 <- rda_Caps_triplot_5_G_RL_P80 +
  annotate(geom="text", x=2, y=1.8, label="R²_adj = 12,94",
              color="black", size = 5)
rda_Caps_triplot_5_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Rel_Tim
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_5_G_RL_P80
dev.off()
```

13.10 Plot RDA's of genus together

```
plot_All_RDA_RL_P80 <- ggarrange(rda_Caps_triplot_3_G_RL_P80, rda_Caps_triplot_2_G_RL_P80,
                                 rda_Caps_triplot_4_G_RL_P80, rda_Caps_triplot_5_G_RL_P80,
                                 heights = c(2,2,2,2),
          labels = c("A","B","C","D"),
          ncol = 1, nrow = 4, align = "v")
plot_All_RDA_RL_P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_AllRDAs_RelAbs_G_
     width=6*dpi,height=15*dpi,res=dpi,
     compression="lzw")
plot_All_RDA_RL_P80
dev.off()
```

#QMP ## Absolute abundances

```
# read in excel with cellcounts
FC <- read.csv2("SHIME_ExtractedCells.csv")

FC_S2 <- dplyr::filter(FC,SHIME ==2)
FC_S2 <- FC_S2[order(FC_S2$Timepoint),]
FC_S3 <- dplyr::filter(FC,SHIME ==3)
FC_S3 <- FC_S3[order(FC_S3$Timepoint),]

FC_S2_T14 <- dplyr::filter(FC_S2,FC_S2$Timepoint == 14, ID == "20190411_S2_D12_D13_D14")
FC_S2 <- dplyr::filter(FC_S2,FC_S2$Timepoint %in% c(2,11,12,18))
FC_S2 <- rbind(FC_S2,FC_S2_T14)
FC_S2 <- FC_S2[order(FC_S2$Timepoint),]

FC_S3 <- dplyr::filter(FC_S3,FC_S3$Timepoint %in% c(2,9,10,12,16))
#
FC <- rbind(FC_S3,FC_S2) #Make sure donor 2 is on top here!!

RelAbs.t <- as.data.frame(t(RelAbs))
  RelAbs.t.meta <- cbind(RelAbs.t, metadata[1:142,])

#select only luminal samples
RelAbs.t.meta.lumen <- dplyr::filter(RelAbs.t.meta,Location == "lumen")

# erase fecal samples from RelAbs
```

```
RelAbsFC <- dplyr::filter(RelAbs.t.meta.lumen,Emulsifier %in% c("Control","SoyLec",
                                                                "RL","P80"))
RelAbsFC <- RelAbsFC[,1:28126]

# multiply abundance data for each sample with Total cellcounts (cfu/ml) for that sample
# RelAbs <- decostand(data.frame(otu.table.t.ns.cpnr[1:319]), 2, method = "total")
# standardize the read counts => calculate relative abundances
# colnames(RelAbs) <- colnames(otu.table.t.ns.cpnr)   #add column names

# Total cellcounts in log's by adding up Intact and Damaged counts
TotalCount <- FC$Intact.cells.conc
RelAbsFC.t <- as.data.frame(t(RelAbsFC))
AbsAbFC.t <- sweep(RelAbsFC,MARGIN =1,TotalCount,'*')
AbsAbFC <- as.data.frame(t(AbsAbFC.t))  #Samples are columns, OTU's are rows

# use this multiplied OTU-table as new dataset for plots
#OTU's are columns, Samples are rows
```

13.11 data wrangling

```
#explore color pallettes
brewer.pal(12, "Paired")

# [1] "#A6CEE3" "#1F78B4" "#B2DF8A" "#33A02C" "#FB9A99" "#E31A1C"
# [7] "#FDBF6F" "#FF7F00" "#CAB2D6" "#6A3D9A" "#FFFF99" "#B15928"

# to display that palette:
display.brewer.pal(12, "Paired")
```

```
brewer.pal(9, "Set1")
display.brewer.pal(12, "Set1")
```

```
#-------------------------------------------------------------------------------------------#
##SAME WORKFLOW AS ABOVE WITH RELATIVE ABUNDANCES
# add taxonomy column
AbsAbFC$Genus <- taxonomy.np.ns$Genus

#summate abundances in each sample per genus
AbsAbFC.Gr <- aggregate(AbsAbFC[1:70], by=list(Category=AbsAbFC$Genus), FUN=sum)

#filter out 8 most abundant genusses
AbsAbFC.Gr$Sums <- rowSums(AbsAbFC.Gr[2:71])   #take rowsums
AbsAbFC.Gr <- AbsAbFC.Gr[order(-AbsAbFC.Gr$Sums),]  #order columns on value of the rowsums
AbsAbFC.Gr.top8 <- AbsAbFC.Gr[1:10,]   #take first 8 Genera's
AbsAbFC.Gr.top8[11,2:72] <- colSums(AbsAbFC.Gr[12:199,2:72])  #add 9th row as a sum of
                                                              #all the rest

#adding "Other" on nineth place of first column
AbsAbFC.Gr.top8$Category <- as.character(AbsAbFC.Gr.top8$Category)
AbsAbFC.Gr.top8$Category[11] <- c("Other")
AbsAbFC.Gr.top8$Category <- as.factor(AbsAbFC.Gr.top8$Category)
```

```
AbsAbFC.Gr.top8$Sums <- NULL  #remove sums column

#rearrange data-table in order to be able to add metadata
AbsAbFC.Gr.top8.R <- tidyr::gather(AbsAbFC.Gr.top8,key = "Samplenr",
                                   value = "AbsoluteAbundance",2:71)

#add metadata
AbsAbFC.merge.gen <- merge(AbsAbFC.Gr.top8.R,metadata, by = "Samplenr")
AbsAbFC.merge.gen <- dplyr::arrange(AbsAbFC.merge.gen,as.numeric(
AbsAbFC.merge.gen$Samplenr))
# AbsAbFC.merge.gen <- unite(AbsAbFC.merge.gen, Treatment, c(EM, EM_Conc), remove=FALSE,
#sep =' - ')

write.csv(AbsAbFC.merge.gen,"Abs_merge_gen.csv")
# AbsAbFC.merge.gen <- readxl::read_xlsx('Abs_merge_gen.xlsx', sheet = "Abs_merge_gen")

AbsAbFC.merge.gen$Category <- as.factor(AbsAbFC.merge.gen$Category)
AbsAbFC.merge.gen$Donor <- as.factor(AbsAbFC.merge.gen$Donor)
AbsAbFC.merge.gen$Timepoint <- as.factor(AbsAbFC.merge.gen$Timepoint)
AbsAbFC.merge.gen$Treatment <- as.factor(AbsAbFC.merge.gen$Treatment)
```

13.12 Plots

13.12.1 General

```
AbsAbFC.merge.gen.lumen <- dplyr::filter(AbsAbFC.merge.gen, Location == 'lumen')
AbsAbFC.merge.gen.lumen$Treatment <- as.factor(AbsAbFC.merge.gen.lumen$Treatment)

X.order <- c("Fecal - 0","Control - 0","P80 - 0.05","P80 - 0.5", "SoyLec - 0.05",
             "SoyLec - 0.5","RL - 0.05","RL - 0.5")

## GENERAL PLOT
barplot.Abs.genus.lumen.Abs <-ggplot(data=AbsAbFC.merge.gen.lumen,
                                     aes(x= Samplenr, y=AbsoluteAbundance,
                                         fill = factor(Category,
                                          levels=as.character(AbsAbFC.Gr.top8$Category))))+
  geom_bar(stat="identity")+labs(fill = "Top 8 Genera")+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+ theme_minimal()+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))
barplot.Abs.genus.lumen.Abs
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotAbsGenusLumen1
     width=12*dpi,height=8*dpi,res=dpi,
     compression="lzw")
barplot.Abs.genus.lumen.Abs
dev.off()

X.order.lumen.Abs <- c("Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - 0","P80 - 0.05 - 1","P80 - 0.05 - 3",
```

```
             "P80 - 0.05 - 7",".","P80 - 0.5 - 0","P80 - 0.5 - 1","P80 - 0.5 - 3",
             "P80 - 0.5 - 7",".","SoyLec - 0.05 - 0","SoyLec - 0.05 - 1",
             "SoyLec - 0.05 - 3", "SoyLec - 0.05 - 7", ".","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.lumen.2.Abs <-ggplot(data=AbsAbFC.merge.gen.lumen,
                                   aes(x= TreatmentTime, y=AbsoluteAbundance,
                                       fill = factor(Category,
                                      levels=as.character(AbsAbFC.Gr.top8$Category)))) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Absolute Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.Abs)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.lumen.2.Abs
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotAbsGenusLumen2
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.lumen.2.Abs
dev.off()

# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
# barplot.genus.lumen.2.Abs
# dev.off()

# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
```

```
# barplot.genus.lumen.2.Abs
# dev.off()
```

13.12.2 Absolute Abundance Lumen RL.P80

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
AbsAbFC.merge.gen.lumen$Category <- as.factor(AbsAbFC.merge.gen.lumen$Category)
AbsAbFC.merge.gen.lumen.RL.P80 <- dplyr::filter(AbsAbFC.merge.gen.lumen,Emulsifier %in%
                                                  c("Control","RL","P80"))

X.order.lumen.RL.P80 <- c("Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - 0","P80 - 0.05 - 1","P80 - 0.05 - 3",
             "P80 - 0.05 - 7",".","P80 - 0.5 - 0","P80 - 0.5 - 1","P80 - 0.5 - 3",
             "P80 - 0.5 - 7",".","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".", "RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3",
             "RL - 0.5 - 7")


barplot.genus.lumen.RL.P80.Abs <-ggplot(data=AbsAbFC.merge.gen.lumen.RL.P80,
                                        aes(x= TreatmentTime, y=AbsoluteAbundance,
                                            fill = factor(Category,
                                       levels=as.character(AbsAbFC.Gr.top8$Category)))) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Absolute Abundance")+
  xlab("Treatment - Time")+
  labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.RL.P80)+
   # scale_fill_brewer(palette="Set1")+
  #scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
                               "#F781BF","#E8D241", "#A65628","#400F73","#FA815F",
                               "#999999","#CD7C5C", "#27AADE","#4EAC77", "#A670CC",
                               "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73",
                               "#FA715F","#997999","#F787BF","#E8D741", "#A65728",
                               "#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
```

```
#axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.lumen.RL.P80.Abs
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotAbsGenusRLP80.
     width=9*dpi,height=6*dpi,res=dpi,compression="lzw")
barplot.genus.lumen.RL.P80.Abs
dev.off()

# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
# barplot.genus.lumen.RL.P80.Abs
# dev.off()
#
# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
# barplot.genus.lumen.RL.P80.Abs
# dev.off()
```

13.12.3 Faecalibacterium lumen

```
##PLOT OTU-1: most abundant Enterobacteriaceae OTU
AbsAbFC.merge.gen.lumen$Category <- as.factor(AbsAbFC.merge.gen.lumen$Category)
AbsAbFC.merge.gen.lumen.Fae <- dplyr::filter(AbsAbFC.merge.gen.lumen,Category ==
                                               "Faecalibacterium" )

X.order.lumen <- c("Fecal - 0 - 0",".",
             "Control - 0 - -7","Control - 0 - 0","Control - 0 - 1","Control - 0 - 3",
             "Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0","P80 - 0.05 - 1",
             "P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7","P80 - 0.5 - 0",
             "P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".", "SoyLec - 0.05 - -7",
             "SoyLec - 0.05 - 0","SoyLec - 0.05 - 1", "SoyLec - 0.05 - 3",
             "SoyLec - 0.05 - 7", ".", "SoyLec - 0.5 - -7","SoyLec - 0.5 - 0",
             "SoyLec - 0.5 - 1","SoyLec - 0.5 - 3","SoyLec - 0.5 - 7",".","RL - 0.05 - -7",
             "RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3","RL - 0.05 - 7", ".",
             "RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1","RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.ABS.lumen.Fae <-ggplot(data=AbsAbFC.merge.gen.lumen.Fae,
                                     aes(x= TreatmentTime, y=AbsoluteAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("Relative Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  #"#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
```

```
  ##4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  #"#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 12))+
  theme(axis.text.y = element_text(size = 12,family='sans',angle=0),
        axis.title.x = element_text(size =14,family='sans',angle=0),
        axis.title.y = element_text(size=14,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
#axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.ABS.lumen.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=9*dpi,height=6*dpi,res=dpi,
     compression="lzw")
barplot.genus.ABS.lumen.Fae
dev.off()
#
# pdf("Plot_Batch_Illumina_Run3_Genera_Cpnr.pdf",width = 12, height = 6, pointsize = 10)
#     barplot.genus.lumen.bil
# dev.off()
#
# png('Plot_Batch_Illumina_Run3_Genera_Cpnr.png', res = 300, width = 5000, height = 2146)
# barplot.genus.lumen.bil
# dev.off()


#P80 & RHAMNOLIPIDS ------------------------------------------------------------------------#
AbsAbFC.merge.gen.lumen.RL.P80$Category <- as.factor(AbsAbFC.merge.gen.lumen.RL.P80$Category)
AbsAbFC.merge.gen.lumen.RL.P80.Fae <- dplyr::filter(AbsAbFC.merge.gen.lumen.RL.P80,Category ==
                                                      "Faecalibacterium" )

X.order.lumen.RL.P80 <- c("Control - 0 - -7","Control - 0 - 0","Control - 0 - 1",
             "Control - 0 - 3","Control - 0 - 7",".","P80 - 0.05 - -7", "P80 - 0.05 - 0",
             "P80 - 0.05 - 1","P80 - 0.05 - 3","P80 - 0.05 - 7",".","P80 - 0.5 - -7",
             "P80 - 0.5 - 0","P80 - 0.5 - 1","P80 - 0.5 - 3","P80 - 0.5 - 7",".",
             "RL - 0.05 - -7","RL - 0.05 - 0","RL - 0.05 - 1","RL - 0.05 - 3",
             "RL - 0.05 - 7", ".","RL - 0.5 - -7","RL - 0.5 - 0","RL - 0.5 - 1",
             "RL - 0.5 - 3","RL - 0.5 - 7")

barplot.genus.ABS.lumen.RL.P80.Fae <-ggplot(data=AbsAbFC.merge.gen.lumen.RL.P80.Fae,
                                            aes(x= TreatmentTime, y=AbsoluteAbundance)) +
  geom_bar(stat="identity")+
  facet_grid(Donor~.,drop = TRUE)+
  theme_minimal()+
```

```
  # ggtitle("Batch Experiments - Barplot Genus")+
  ylab("ABsolure Abundance")+
  xlab("Treatment - Time")+
  # labs(fill = "Top 10 Genera")+
  scale_x_discrete(limits=X.order.lumen.RL.P80)+
   # scale_fill_brewer(palette="Set1")+
  # scale_fill_manual(values = c("#D6456CFF", "#FEB37BFF", "#6B1D81FF", "#160F3BFF",
  #"#FA815F", "#FCFDBF", "#AB337C", "#400F73", "#999999"))+
  # scale_fill_manual(values = c("#CD5C5C", "#21AADE","#4EAC57", "#A660CC", "#E1A13F",
  #"#F781BF","#E8D241", "#A65628","#400F73","#FA815F","#999999","#CD7C5C", "#27AADE",
  #"#4EAC77", "#A670CC", "#E1A17F", "#F787BF","#E8D741", "#A65728","#407F73","#FA715F",
  #"#997999","#F787BF","#E8D741", "#A65728","#407F73","#FA715F","#997999"))+
  theme(axis.text.x=element_text(angle=90, hjust=1, vjust = 0,size = 8))+
  theme(axis.text.y = element_text(size = 10,family='sans',angle=0),
        axis.title.x = element_text(size =12,family='sans',angle=0),
        axis.title.y = element_text(size=12,family='sans',angle=90),
        legend.text = element_text(size=10,family='sans',angle=0, face = "italic"),
        legend.title = element_text(size=10,family='sans',angle=0),
        strip.text = element_text(size=10,angle=0,family='sans'),
        legend.position = "bottom",
        legend.key.size = unit(0.3,"cm"),
        panel.grid.major=element_blank(),
        panel.grid.minor=element_blank())+
  theme(axis.line.y=element_line(colour="black"),axis.line=element_line(colour="black"))
  # theme(axis.text.x=element_text(angle=90, hjust=1, size = 6),
# axis.text.y=element_text(size = 6) )+
      #theme(axis.text.x = element_text(size=10), legend.title=element_text(size=10),
#legend.text=element_text(size=7))
barplot.genus.ABS.lumen.RL.P80.Fae
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_BarplotGenusFaecaliba
     width=7*dpi,height=5*dpi,
     res=dpi,compression="lzw")
barplot.genus.ABS.lumen.RL.P80.Fae
dev.off()
```

#RDA - RL - TWEEN80 ## Genus - global

```
metadata.lumen <- dplyr::filter(metadata,Location == "lumen")
metadata.Abs <- dplyr::filter(metadata.lumen,Emulsifier %in%
                                c("Control","SoyLec","RL","P80")) # no fecal samples
AbsAb.Gr.meta <- cbind(t(AbsAbFC.Gr[2:71]),metadata.Abs)

##split RelAbs by donor
AbsAb.Gr.meta.RL.P80 <- dplyr::filter(AbsAb.Gr.meta,Emulsifier %in%
                                        c("Control","P80","RL"))

## partial distance based RDA
Xvars.rda.G.RL.P80 <- AbsAb.Gr.meta.RL.P80[199:209]
Yvars.rda.G.RL.P80 <- as.matrix(AbsAb.Gr.meta.RL.P80[1:199])

#-------------------------------------------------------------------------------------#
rda_G_Abs1_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Donor + Treatment + Timepoint,
```

```
                              data = Xvars.rda.G.RL.P80, distance = "jaccard")
anova(rda_G_Abs1_RL_P80 , by = "term", permutations = 199)
  #P_Donor = 0,005 **
  #P_Treatment = 0,005 **
  #P_Timepoint = 0,005 **

# plot(rda_G_Abs1_RL_P80)
# points(rda_G_Abs1_RL_P80, pch = 19, col = colvec[Xvars.rda.G.RL.P80$Donor])
# text(rda_G_Abs1_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda.G.RL.P80$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Abs1_RL_P80 <- summary(rda_G_Abs1_RL_P80)
  #CAP1 = 22,93%
  #CAP2 = 9,76%

species1G_Abs_RL_80 <- data.frame(smry_rda_G_Abs1_RL_P80$species[,1:2])  #we only need
                                  #the first 2 coordinates for both species and sites.
species1G_Abs_RL_80$Genus <- AbsAbFC.Gr[,1]

sites1G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs1_RL_P80$sites[,1:2])
sites.meta1G_Abs_RL_P80 <- cbind(sites1G_Abs_RL_P80, AbsAb.Gr.meta.RL.P80[199:209])
centroids1G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs1_RL_P80$centroids[,1:2])
rownames(centroids1G_Abs_RL_P80) <- c("D1","D2","Control - 0","P80 - 0.05","P80 - 0.5",
                                      "RL - 0.05","RL - 0.5","T2","T9","T10","T11",
                                      "T12","T14","T16","T18")
# biplot <- data.frame(smry_rda_OTU_Rel1$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
sites.meta1G_Abs_RL_P80$Emulsifier <- as.factor(sites.meta1G_Abs_RL_P80$Emulsifier)
sites.meta1G_Abs_RL_P80$Donor <- as.factor(sites.meta1G_Abs_RL_P80$Donor)

# Colored by treatment
rda_Caps_plot_Abs_Treatment_G_RL_80 <- ggplot(sites.meta1G_Abs_RL_P80,
                                              aes(x=CAP1, y=CAP2, colour = Treatment)) +
  geom_point(size = 6) +
  geom_text(data = sites.meta1G_Abs_RL_P80, aes(x= CAP1, y = CAP2,  label=Timepoint,
                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)))) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("Black","#d28eed", "#844da8" ,"#74dff7","#125087",
                               "#f5e2a2", "#e3d029", "#F781BF","#f2b385","#d97914"))
rda_Caps_plot_Abs_Treatment_G_RL_80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Abs_Treatment_G_RL_80  <- rda_Caps_plot_Abs_Treatment_G_RL_80 +
  geom_segment(data=species1G_Abs_RL_80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G_Abs_RL_80[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
```

```
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Abs_Treatment_G_RL_80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Abs_Treatment_G_RL_80  <- rda_Caps_biplot_Abs_Treatment_G_RL_80 +
  geom_segment(data=centroids1G_Abs_RL_P80[1:6,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G_Abs_RL_P80[1:6,],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G_Abs_RL_P80[1:6,]),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (9,76%)") +
  xlab("CAP1 (22,93%)")+
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))
rda_Caps_triplot1_Abs_Treatment_G_RL_80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Abs_Col
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Abs_Treatment_G_RL_80
dev.off()

# Colored by Donor
rda_Caps_plot_Donor_G_Abs_RL_80 <- ggplot(sites.meta1G_Abs_RL_P80,
                                          aes(x=CAP1, y=CAP2, colour = Donor)) +
  geom_point() +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
 scale_color_manual(values = c("#125087", "#d28eed", "#f5e2a2","#F781BF","#f2b385",
                               "#844da8","#74dff7","Black","#999999","#e3d029","#d97914"))
rda_Caps_plot_Donor_G_Abs_RL_80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_Donor_G_Abs_RL_80  <- rda_Caps_plot_Donor_G_Abs_RL_80 +
  geom_segment(data=species1G_Abs_RL_80[1:10,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species1G_Abs_RL_80[1:10,],
            aes(x=CAP1,y=CAP2, label=Genus[1:10],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_Donor_G_Abs_RL_80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot1_Donor_G_Abs_RL_80  <- rda_Caps_biplot_Donor_G_Abs_RL_80 +
  geom_segment(data=centroids1G_Abs_RL_P80[c(1,2),], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids1G_Abs_RL_P80[c(1,2),],
            aes(x=CAP1,y=CAP2, label=rownames(centroids1G_Abs_RL_P80[c(1,2),]),
```

```
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
  ylab("CAP2 (9,76%)") +
  xlab("CAP1 (22,93%)")+
  theme_bw()+ theme(axis.title.x=element_text( size = 16),
                    axis.title.y=element_text(size = 16),
                    legend.text = element_text(size = 14),
                    legend.title = element_text(size = 16))
rda_Caps_triplot1_Donor_G_Abs_RL_80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Donor_G_
     width = 3000,height = 2500,res =300)
rda_Caps_triplot1_Donor_G_Abs_RL_80
dev.off()
```

13.13 Genus - Treatment

```
## RDA based with all but  Donor conditioned out
rda_G_Abs3_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Treatment + Condition(Donor) +
                                Condition(Timepoint), data = Xvars.rda.G.RL.P80,
                              distance = "jaccard")
anova(rda_G_Abs3_RL_P80, by = "term", permutations = 199)
  #significance Treatment = 0.005 *

# plot(rda_G_Abs3_RL_P80)
# points(rda_G_Abs3_RL_P80, pch = 19, col = colvec[Xvars.rda.G.RL.P80$Donor])
# text(rda_G_Abs3_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda.G.RL.P80$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Abs3_RL_P80 <- summary(rda_G_Abs3_RL_P80)
  #CAP1 = 11,09%   11.05
  #CAP2 = 4,35%    4.44

Runadj_G_Rel3_RL_P80<- vegan::RsquareAdj(rda_G_Abs3_RL_P80)$r.squared  #=> 11.92
Radj_G_Rel3_RL_P80 <- RsquareAdj(rda_G_Abs3_RL_P80)$adj.r.squared    #=>  8.33

species3G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs3_RL_P80$species[,1:2])  #we only need
                                   #the first 2 coordinates for both species and sites.
species3G_Abs_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites3G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs3_RL_P80$sites[,1:2])
sites.meta3G_Abs_RL_P80 <- cbind(sites3G_Abs_RL_P80, AbsAb.Gr.meta.RL.P80[199:209])
centroids3G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs3_RL_P80$centroids[,1:2])
rownames(centroids3G_Abs_RL_P80) <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05",
                                      "RL - 0.5")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
```

```
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta3G_Abs_RL_P80$Donor <- as.factor(sites.meta3G_Abs_RL_P80$Donor)

rda_Caps_plot_3_Abs_G_RL_P80 <- ggplot(sites.meta3G_Abs_RL_P80,
                                       aes(x=CAP1, y=CAP2, colour = Treatment)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999", "#d28eed", "#844da8" ,"#74dff7","#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385", "#d97914"))
rda_Caps_plot_3_Abs_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_3_Abs_G_RL_P80 <- rda_Caps_plot_3_Abs_G_RL_P80 +
  # geom_text(data = sites.meta3G_Abs_RL_P80, aes(x= CAP1, y = CAP2,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)))) +
  geom_segment(data=species3G_Abs_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species3G_Abs_RL_P80[1:5,],
            aes(x=CAP1,y=CAP2, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_3_Abs_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_3_Abs_G_RL_P80 <- rda_Caps_biplot_3_Abs_G_RL_P80 +
  # geom_segment(data=centroids3G_Abs_RL_P80, aes(x=0, xend=CAP1, y=0, yend=CAP2),
  #                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids3G_Abs_RL_P80,
            aes(x=CAP1,y=CAP2, label=rownames(centroids3G_Abs_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
   ylab("CAP2 (4,44%)") +
   xlab("CAP1 (11,05%)")+
   xlim(-3,3)+
   ylim(-3,3)+
   theme_bw()+ theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_3_Abs_G_RL_P80
```

```
rda_Caps_triplot_3_Abs_G_RL_P80 <- rda_Caps_triplot_3_Abs_G_RL_P80 +
  annotate(geom="text", x=2, y=2.5, label="R²_adj = 8,33",
              color="black", size = 5)
rda_Caps_triplot_3_Abs_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Abs_Trea
     width = 3000,height = 2500,res =300)
```

```
rda_Caps_triplot_3_Abs_G_RL_P80
dev.off()
```

13.14 Genus - Donor

```
## RDA based with all but  Donor conditioned out
rda_G_Abs2_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Donor + Condition(Treatment) +
                                Condition(Timepoint), data = Xvars.rda.G.RL.P80,
                              distance = "jaccard")
anova(rda_G_Abs2_RL_P80, by = "term", permutations = 199)
  #significance Treatment = 0.005 **

# plot(rda_G_Abs2_RL_P80)
# points(rda_G_Abs2_RL_P80, pch = 19, col = colvec[Xvars.rda.G.RL.P80$Donor])
# text(rda_G_Abs2_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda.G.RL.P80$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Abs2_RL_P80 <- summary(rda_G_Abs2_RL_P80)
  #CAP1 = 13.72
  #MDS1 = 15.96

Runadj_G_Abs2_RL_P80<- vegan::RsquareAdj(rda_G_Abs2_RL_P80)$r.squared  #=> 7.46%
Radj_G_Abs2_RL_P80 <- RsquareAdj(rda_G_Abs2_RL_P80)$adj.r.squared    #=> 8.02%

species2G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs2_RL_P80$species[,1:2])  #we only need
                                    #the first 2 coordinates for both species and sites.
species2G_Abs_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites2G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs2_RL_P80$sites[,1:2])
sites.meta2G_Abs_RL_P80 <- cbind(sites2G_Abs_RL_P80, AbsAb.Gr.meta.RL.P80[199:209])
centroids2G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs2_RL_P80$centroids[,1:2])
rownames(centroids2G_Abs_RL_P80) <- c("Donor 1","Donor 2")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta2G_Abs_RL_P80$Donor <- as.factor(sites.meta2G_Abs_RL_P80$Donor)

rda_Caps_plot_2_Abs_G_RL_P80 <- ggplot(sites.meta2G_Abs_RL_P80,
                                       aes(x=CAP1, y=MDS1, colour = Donor)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999", "#d28eed", "#844da8" ,"#74dff7","#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385","#d97914"))
rda_Caps_plot_2_Abs_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_2_Abs_G_RL_P80 <- rda_Caps_plot_2_Abs_G_RL_P80 +
  # geom_text(data = sites.meta2G_Abs_RL_P80, aes(x= CAP1, y = MDS1,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(MDS1)))) +
  geom_segment(data=species2G_Abs_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=MDS1),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species2G_Abs_RL_P80[1:5,],
            aes(x=CAP1,y=MDS1, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_2_Abs_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_2_Abs_G_RL_P80 <- rda_Caps_biplot_2_Abs_G_RL_P80 +
  # geom_segment(data=centroids2G_Abs_RL_P80, aes(x=0, xend=CAP1, y=0, yend=MDS1),
  #                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids2G_Abs_RL_P80,
            aes(x=CAP1,y=MDS1, label=rownames(centroids2G_Abs_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(MDS1))),
            color="blue", size=4)+
   ylab("MDS1 (15,96%)") +
   xlab("CAP1 (13,72%)")+
   xlim(-2.5,2)+
   ylim(-2,2)+
   theme_bw()+ theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_2_Abs_G_RL_P80
```

```
rda_Caps_triplot_2_Abs_G_RL_P80 <- rda_Caps_triplot_2_Abs_G_RL_P80 +
  annotate(geom="text", x=1, y=1.5, label="R²_adj = 8,02",
              color="black", size = 5)
rda_Caps_triplot_2_Abs_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Abs_Don
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_2_Abs_G_RL_P80
dev.off()
```

13.15 Genus - Timepoint

```
## RDA based with all but  Donor conditioned out
rda_G_Abs4_RL_P80 <- capscale(Yvars.rda.G.RL.P80 ~ Timepoint + Condition(Donor) +
                                Condition(Treatment), data = Xvars.rda.G.RL.P80,
                              distance = "jaccard")
anova(rda_G_Abs4_RL_P80, by = "term", permutations = 199)
  #significance Treatment = 0.005 **

# plot(rda_G_Abs3_RL_P80)
```

```
# points(rda_G_Abs3_RL_P80, pch = 19, col = colvec[Xvars.rda.G.RL.P80$Donor])
# text(rda_G_Abs3_RL_P80, display = "species", cex = 0.8, col = "red")
# legend("bottomright", legend = levels(Xvars.rda.G.RL.P80$Donor), col = colvec, pch = 19,
#        bty = "n")

#-------------------------------------------------------------------------------------#
#plotting of NGS-RDA using ggplot
smry_rda_G_Abs4_RL_P80 <- summary(rda_G_Abs4_RL_P80)
  #CAP1 = 14.48%
  #CAP2 = 11.44%

Runadj_G_Rel4_RL_P80<- vegan::RsquareAdj(rda_G_Abs4_RL_P80)$r.squared  #=> 24.57%
Radj_G_Rel4_RL_P80 <- RsquareAdj(rda_G_Abs4_RL_P80)$adj.r.squared    #=> 17.70%

species4G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs4_RL_P80$species[,1:2])  #we only need
                                    #the first 2 coordinates for both species and sites.
species4G_Abs_RL_P80$Genus <- otu.table.t.ns.cpnr.relabun.Gr[,1]

sites4G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs4_RL_P80$sites[,1:2])
sites.meta4G_Abs_RL_P80 <- cbind(sites4G_Abs_RL_P80, AbsAb.Gr.meta.RL.P80[199:209])
centroids4G_Abs_RL_P80 <- data.frame(smry_rda_G_Abs4_RL_P80$centroids[,1:2])
rownames(centroids4G_Abs_RL_P80) <- c("T2","T9","T10","T11","T12","T14","T16","T18")
# biplot <- data.frame(smry_rda_OTU_Caps$centroids)     = same as centroids
#first plot the sites:, i.e. all the samples.(319)
# sites.meta3$EM <- factor(sites.meta2$EM, levels = c("CMC","P80","SoyLec","SL","RL",
#"Control",))
# colnames(sites.meta2)[7] <- c("Emulsifier")
sites.meta4G_Abs_RL_P80$Donor <- as.factor(sites.meta4G_Abs_RL_P80$Donor)

rda_Caps_plot_4_Abs_G_RL_P80 <- ggplot(sites.meta4G_Abs_RL_P80,
                                       aes(x=CAP1, y=CAP2, colour = Timepoint)) +
  geom_point(size = 2) +
  #geom_text(aes(label=rownames(sites)),size=4) +
  geom_hline(yintercept=0, linetype="dotted") +
  geom_vline(xintercept=0, linetype="dotted") +
  labs(fill = "Emulsifier") +
  scale_color_manual(values = c("#999999","#d28eed","#844da8","#74dff7","#125087",
                                "#f5e2a2", "#e3d029", "#F781BF","#f2b385", "#d97914"))
rda_Caps_plot_4_Abs_G_RL_P80
```

```
#now we plot the arrow that indicate the species, i.e. the OTU's in this case
#we'll choose to only plot the first 10 or so OTU's
rda_Caps_biplot_4_Abs_G_RL_P80 <- rda_Caps_plot_4_Abs_G_RL_P80 +
  # geom_text(data = sites.meta4G_Abs_RL_P80, aes(x= CAP1, y = CAP2,  label=Timepoint,
  #                hjust=1*(1-sign(CAP1)),vjust=1*(1-sign(CAP2)))) +
  geom_segment(data=species4G_Abs_RL_P80[1:5,], aes(x=0, xend=CAP1, y=0, yend=CAP2),
               color="red", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=species4G_Abs_RL_P80[1:5,],
            aes(x=CAP1,y=CAP2, label=Genus[1:5],
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="red", size=4)+
  theme_bw()
rda_Caps_biplot_4_Abs_G_RL_P80
```

```
#now to makeit a triplot, we add the centroids as well, i.e. the explanatory variables
rda_Caps_triplot_4_Abs_G_RL_P80 <- rda_Caps_biplot_4_Abs_G_RL_P80 +
  geom_segment(data=centroids4G_Abs_RL_P80, aes(x=0, xend=CAP1, y=0, yend=CAP2),
                color="blue", arrow=arrow(length=unit(0.01,"npc"))) +
  geom_text(data=centroids4G_Abs_RL_P80,
            aes(x=CAP1,y=CAP2, label=rownames(centroids4G_Abs_RL_P80),
                hjust=0.5*(1-sign(CAP1)),vjust=0.5*(1-sign(CAP2))),
            color="blue", size=4)+
   ylab("CAP2 (11,44%)") +
   xlab("CAP1 (14,48%)")+
   xlim(-2.5,2)+
   ylim(-2,2)+
   theme_bw()+ theme(axis.title.x=element_text( size = 16),
                     axis.title.y=element_text(size = 16),
                     legend.text = element_text(size = 14),
                     legend.title = element_text(size = 16))
rda_Caps_triplot_4_Abs_G_RL_P80
```

```
rda_Caps_triplot_4_Abs_G_RL_P80 <- rda_Caps_triplot_4_Abs_G_RL_P80 +
  annotate(geom="text", x=1, y=1.5, label="R²_adj = 17,70",
              color="black", size = 5)
rda_Caps_triplot_4_Abs_G_RL_P80
```

```
dpi=300
tiff(file ="/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_RDA_Abs_Tim
     width = 3000,height = 2500,res =300)
rda_Caps_triplot_4_Abs_G_RL_P80
dev.off()
```

13.16 Plot RDA's of genus together

```
plot_All_RDA_Abs_RL_P80 <- ggarrange(rda_Caps_triplot_3_Abs_G_RL_P80,
                                     rda_Caps_triplot_2_Abs_G_RL_P80,
                                     rda_Caps_triplot_4_Abs_G_RL_P80, heights = c(2,2,2),
          labels = c("E","F","G"),
          ncol = 1, nrow = 3, align = "v")
plot_All_RDA_Abs_RL_P80
```

```
dpi=300
tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_NGS_AllRDAs_AbsAbs_G_
     width=6*dpi,height=12*dpi,res=dpi,
     compression="lzw")
plot_All_RDA_Abs_RL_P80
dev.off()
```

14 Deseq - RL -P80

```
# pre-processing data
minobs=1
DeseqOTU <- otu.table.t.ns.cpnr[1:142]
#create genus level OTU-table.
```

```
#filter taxonomytable with OTU numbers retained after wnwn filtering
rownames(taxonomy.np.ns) <-gsub("Otu0+([1,2,3,4,5,6,7,8,9]+)","OTU\\1",
                                rownames(taxonomy.np.ns))
rownames(DeseqOTU) <-gsub("Otu0+([1,2,3,4,5,6,7,8,9]+)","OTU\\1",rownames(DeseqOTU))
genus.table.tax <- taxonomy.np.ns[rownames(taxonomy.np.ns) %in% rownames(DeseqOTU),]

# stick genuslevel to OTU-table
genus.tax <- cbind(genus.table.tax$Genus,DeseqOTU)

#aggregate to obtain genus-level count-table
genus.table <- aggregate(genus.tax[2:143], by =
                           list(Category = genus.tax$`genus.table.tax$Genus`),
                         FUN = sum)
rownames(genus.table) <- genus.table$Category
genus.table <- genus.table[,-1]

# Split up data bewteen Lumen and Mucus and filter out treatment-timepoints
genus.table <- as.data.frame(t(genus.table))
genus.table.meta <- cbind(genus.table, metadata[1:142,])

#RL P80
genus.table.meta.RL.P80 <- dplyr::filter(genus.table.meta,Emulsifier %in% c("RL",
                                                                             "Control",
                                                                             "P80"))

genus.table.meta.RL.P80.lumen <- dplyr::filter(genus.table.meta.RL.P80,Location == "lumen")
genus.table.meta.RL.P80.mucus <- dplyr::filter(genus.table.meta.RL.P80,Location == "mucin")

genus.table.meta.RL.P80.lumen.treat <-dplyr::filter(genus.table.meta.RL.P80.lumen,TimeCode
                                                    %in% c("1","3","7"))
genus.table.meta.RL.P80.mucus.treat <-dplyr::filter(genus.table.meta.RL.P80.mucus,TimeCode
                                                    %in% c("3","5","7"))
```

14.1 LUMEN

```
# deseq-analysis LUMEN
genus.table.wnwn.lumen.treat <- genus.table.meta.RL.P80.lumen.treat[1:199]
metadata.lumen.treat <- genus.table.meta.RL.P80.lumen.treat[200:209]

genus.table.wnwn.lumen.treat <- as.data.frame(t(genus.table.wnwn.lumen.treat))
deseqdata.genus = as.matrix(genus.table.wnwn.lumen.treat+1)
deseqdata.genus <- DESeqDataSetFromMatrix(deseqdata.genus,colData=metadata.lumen.treat,
                                          design= ~ TimeCode + Donor + Treatment)

#normalization
deseqdata.genus <- estimateSizeFactors(deseqdata.genus)
sizeFactors(deseqdata.genus)

testdiff.genus.lumen <- DESeq(deseqdata.genus,test = "LRT", reduced = ~Donor + TimeCode,
                              parallel = TRUE)
restestdiff.genus.lumen <- results(testdiff.genus.lumen)
```

```
#restestdiffordered.genus <- restestdiff[order(restestdiff.genus$padj),]
summary(restestdiff.genus.lumen)
 #7.5%  (15) upregulated
 #28%  (56) downregulated

# Pairwise significant differences of all treatments vs Control.
  ## P80_0,05 vs control
resP80_0.05_contr_lumen_G <- results(testdiff.genus.lumen,
                                     contrast=c("Treatment","P80 - 0.05","Control - 0"),
                                     test="Wald",alpha=0.05,independentFiltering=FALSE)
  #shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resP80_0.05_contr_lumen_G <- lfcShrink(testdiff.genus.lumen,
                                       contrast=c("Treatment","P80 - 0.05","Control - 0"),
                                       res=resP80_0.05_contr_lumen_G, type = "ashr")
  #extract significantly different OTU's
resP80_0.05_contrsign_lumen_G <- resP80_0.05_contr_lumen_G$padj < 0.05
  #resSoyLec_0.05_contrsign_G <- lfcShrink(testdiff.genus.lumen,
#contrast=c("Treatment","SoyLec_0.05","Control_0"),res=resSoyLec_0.05_contr_G,
#type = "ashr")$padj<0.05
resP80_0.05_contrsignOTU_lumen_G <- rownames(genus.table.wnwn.lumen.treat)[
  resP80_0.05_contrsign_lumen_G==TRUE]

resP80_0.05_contrvol_lumen_G <- volcanoLisa(resP80_0.05_contr_lumen_G,
                                            taxlevel="genus",1,-1)+
  ggtitle("P80-0,05% \u2013 Control \u2013 Lumen") +
  theme(legend.box="vertical")
resP80_0.05_contrvol_lumen_G
```

```
mylegend_volcano_P80_0.05_contr_G <- get_legend(resP80_0.05_contrvol_lumen_G)
resP80_0.05_contrvol_lumen_G <- papertheme(resP80_0.05_contrvol_lumen_G) +
  theme(legend.position="none",axis.line = element_line(color="black"))+
  scale_y_continuous(limits = c(0,20))
resP80_0.05_contrvol_lumen_G
```

```
 ## P80_0,5 vs control
resP80_0.5_contr_lumen_G <- results(testdiff.genus.lumen,
                                    contrast=c("Treatment","P80 - 0.5","Control - 0"),
                                    test="Wald",alpha=0.05,independentFiltering=FALSE)
  #shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resP80_0.5_contr_lumen_G <- lfcShrink(testdiff.genus.lumen,
                                      contrast=c("Treatment","P80 - 0.5","Control - 0"),
                                      res=resP80_0.5_contr_lumen_G, type = "ashr")
  #extract significantly different OTU's
resP80_0.5_contrsign_lumen_G <- resP80_0.5_contr_lumen_G$padj <0.05
  #resP80_0.5_contrsign_G <- lfcShrink(testdiff.genus.lumen,
# contrast=c("Treatment","P80_0.5","Control_0"),
# res=resP80_0.5_contr_G,type = "ashr")$padj<0.05
resP80_0.5_contrsignOTU_lumen_G <- rownames(genus.table.wnwn.lumen.treat)[
  resP80_0.5_contrsign_lumen_G==TRUE]

resP80_0.5_contrvol_lumen_G <- volcanoLisa(resP80_0.5_contr_lumen_G,
                                           taxlevel="genus",1,-1) +
  ggtitle("P80-0,5% \u2013 Control \u2013 Lumen") +
  theme(legend.box="vertical")
```

```
resP80_0.5_contrvol_lumen_G
```

```
mylegend_volcano_P80_0.5_contr_lumen_G <- get_legend(resP80_0.5_contrvol_lumen_G)
resP80_0.5_contrvol_lumen_G <- papertheme(resP80_0.5_contrvol_lumen_G) +
  theme(legend.position="none",axis.line = element_line(color="black"))+
  scale_y_continuous(limits = c(0,20))
resP80_0.5_contrvol_lumen_G
```

```
  ## RL_0,05 vs control
resRL_0.05_contr_lumen_G <- results(testdiff.genus.lumen,
                                    contrast=c("Treatment","RL - 0.05","Control - 0"),
                                    test="Wald",alpha=0.05,independentFiltering=FALSE)
  #shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resRL_0.05_contr_lumen_G <- lfcShrink(testdiff.genus.lumen,
                                      contrast=c("Treatment","RL - 0.05","Control - 0"),
                                      res=resRL_0.05_contr_lumen_G, type = "ashr")
  #extract significantly different OTU's
resRL_0.05_contrsign_lumen_G <- resRL_0.05_contr_lumen_G$padj < 0.05
  #resRL_0.05_contrsign_G <- lfcShrink(testdiff.genus.lumen,
#contrast=c("Treatment","RL_0.05","Control_0"),res=resRL_0.05_contr_G,type="ashr")$padj<0.05
resRL_0.05_contrsignOTU_lumen_G <- rownames(genus.table.wnwn.lumen.treat)[
  resRL_0.05_contrsign_lumen_G==TRUE]

resRL_0.05_contrvol_lumen_G <- volcanoLisa(resRL_0.05_contr_lumen_G,
                                           taxlevel="genus",1,-1)+
  ggtitle("RL-0,05% \u2013 Control \u2013 Lumen") +  theme(legend.box="vertical")
resRL_0.05_contrvol_lumen_G
```

```
mylegend_volcano_RL_0.05_contr_lumen_G <- get_legend(resRL_0.05_contrvol_lumen_G)
resRL_0.05_contrvol_lumen_G <- papertheme(resRL_0.05_contrvol_lumen_G) +
  theme(legend.position="none",axis.line = element_line(color="black"))+
  scale_y_continuous(limits = c(0,30))
resRL_0.05_contrvol_lumen_G
```

```
 ## RL_0,5 vs control
resRL_0.5_contr_lumen_G <- results(testdiff.genus.lumen,
                                   contrast=c("Treatment","RL - 0.5","Control - 0"),
                                   test="Wald",alpha=0.05,independentFiltering=FALSE )
  #shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resRL_0.5_contr_lumen_G <- lfcShrink(testdiff.genus.lumen,
                                     contrast=c("Treatment","RL - 0.5","Control - 0"),
                                     res=resRL_0.5_contr_lumen_G, type = "ashr")
  #extract significantly different OTU's
resRL_0.5_contrsign_lumen_G <- resRL_0.5_contr_lumen_G$padj <0.05
  #resRL_0.5_contrsign_G <- lfcShrink(testdiff.genus.lumen,
#contrast=c("Treatment","RL_0.5","Control_0"),res=resRL_0.5_contr_G,type = "ashr")$padj<0.05
resRL_0.5_contrsignOTU_lumen_G <- rownames(genus.table.wnwn.lumen.treat)[
  resRL_0.5_contrsign_lumen_G==TRUE]

resRL_0.5_contrvol_lumen_G <- volcanoLisa(resRL_0.5_contr_lumen_G,
                                          taxlevel="genus",1,-1) +
  ggtitle("RL-0,5% \u2013 Control \u2013 Lumen") +
  theme(legend.box="vertical")
resRL_0.5_contrvol_lumen_G
```

```
mylegend_volcano_RL_0.5_contr_lumen_G <- get_legend(resRL_0.5_contrvol_lumen_G)
resRL_0.5_contrvol_lumen_G <- papertheme(resRL_0.5_contrvol_lumen_G) +
  theme(legend.position="none",axis.line = element_line(color="black"))+
  scale_y_continuous(limits = c(0,30))
resRL_0.5_contrvol_lumen_G
```

14.2 MUCUS

```
### deseq-analysis MUCUS
genus.table.wnwn.mucus.treat <- genus.table.meta.RL.P80.mucus.treat[1:199]
metadata.mucus.treat <- genus.table.meta.RL.P80.mucus.treat[200:209]

genus.table.wnwn.mucus.treat <- as.data.frame(t(genus.table.wnwn.mucus.treat))
deseqdata.genus = as.matrix(genus.table.wnwn.mucus.treat+1)
deseqdata.genus <- DESeqDataSetFromMatrix(deseqdata.genus,
                                          colData=metadata.mucus.treat,
                                          design= ~ TimeCode + Donor + Treatment)

  #normalization
deseqdata.genus <- estimateSizeFactors(deseqdata.genus)
sizeFactors(deseqdata.genus)

testdiff.genus.mucus <- DESeq(deseqdata.genus,test = "LRT",
                              reduced = ~Donor + TimeCode, parallel = TRUE)
restestdiff.genus.mucus <- results(testdiff.genus.mucus)

#restestdiffordered.genus <- restestdiff[order(restestdiff.genus$padj),]
summary(restestdiff.genus.mucus)
 #5%  (10) upregulated
 #30% (60)downregulated
# Volcano plots sign genera

# Pairwise significant differences of all treatments vs Control.
## P80_0,05 vs control
resP80_0.05_contr_mucus_G <- results(testdiff.genus.mucus,
                                     contrast=c("Treatment","P80 - 0.05","Control - 0"),
                                     test="Wald",
                                     alpha=0.05,independentFiltering=FALSE)
#shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resP80_0.05_contr_mucus_G <- lfcShrink(testdiff.genus.mucus,
                                       contrast=c("Treatment","P80 - 0.05","Control - 0"),
                                       res=resP80_0.05_contr_mucus_G,type = "ashr")
#extract significantly different OTU's
resP80_0.05_contrsign_mucus_G <- resP80_0.05_contr_mucus_G$padj < 0.05
#resP80_0.05_contrsign_G <- lfcShrink(testdiff.genus.mucus,
#contrast=c("Treatment","P80_0.05","Control_0"),res=resP80_0.05_contr_G,
#type = "ashr")$padj<0.05
resP80_0.05_contrsignOTU_mucus_G <- rownames(genus.table.wnwn.mucus.treat)[
  resP80_0.05_contrsign_mucus_G==TRUE]

resP80_0.05_contrvol_mucus_G <- volcanoLisa(resP80_0.05_contr_mucus_G,
                                            taxlevel="genus",1,-1) +
```

```
  ggtitle("P80-0,05% \u2013 Control \u2013 Mucus") + theme(legend.box="vertical")
#plotting volcanoplot
resP80_0.05_contrvol_mucus_G
```

```
mylegend_volcano_P80_0.05_contr_mucus_G <- get_legend(resP80_0.05_contrvol_mucus_G)
resP80_0.05_contrvol_mucus_G <- papertheme(resP80_0.05_contrvol_mucus_G) +
    theme(legend.position="none",axis.line = element_line(color="black"))+
    scale_y_continuous(limits = c(0,20))
resP80_0.05_contrvol_mucus_G
```

```
## P80_0,5 vs control
resP80_0.5_contr_mucus_G <- results(testdiff.genus.mucus,
                                    contrast=c("Treatment","P80 - 0.5","Control - 0"),
                                    test="Wald",alpha=0.05,independentFiltering=FALSE)
#shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resP80_0.5_contr_mucus_G <- lfcShrink(testdiff.genus.mucus,
                                      contrast=c("Treatment","P80 - 0.5","Control - 0"),
                                      res=resP80_0.5_contr_mucus_G, type = "ashr")
#extract significantly different OTU's
resP80_0.5_contrsign_mucus_G <- resP80_0.5_contr_mucus_G$padj <0.05
#resP80_0.5_contrsign_G <- lfcShrink(testdiff.genus,
#contrast=c("Treatment","P80_0.5","Control_0"),
#res=resP80_0.5_contr_G,type = "ashr")$padj<0.05
resP80_0.5_contrsignOTU_mucus_G <- rownames(genus.table.wnwn.mucus.treat)[
    resP80_0.5_contrsign_mucus_G==TRUE]
resP80_0.5_contrsignOTU_mucus_G <- na.omit(resP80_0.5_contrsignOTU_mucus_G)

resP80_0.5_contrvol_mucus_G <- volcanoLisa(resP80_0.5_contr_mucus_G,
                                           taxlevel="genus",1,-1) +
    ggtitle("P80-0,5% \u2013 Control \u2013 Mucus") +  theme(legend.box="vertical")
resP80_0.5_contrvol_mucus_G
```

```
mylegend_volcano_P80_0.5_contr_mucus_G <- get_legend(resP80_0.5_contrvol_mucus_G)
resP80_0.5_contrvol_mucus_G <- papertheme(resP80_0.5_contrvol_mucus_G) +
    theme(legend.position="none",axis.line = element_line(color="black"))+
    scale_y_continuous(limits = c(0,20))
resP80_0.5_contrvol_mucus_G
```

```
## RL_0,05 vs control
resRL_0.05_contr_mucus_G <- results(testdiff.genus.mucus,
                                    contrast=c("Treatment","RL - 0.05","Control - 0"),
                                    test="Wald",alpha=0.05,independentFiltering=FALSE)
#shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resRL_0.05_contr_mucus_G <- lfcShrink(testdiff.genus.mucus,
                                      contrast=c("Treatment","RL - 0.05","Control - 0"),
                                      res=resRL_0.05_contr_mucus_G,type = "ashr")
#extract significantly different OTU's
resRL_0.05_contrsign_mucus_G <- resRL_0.05_contr_mucus_G$padj < 0.05
#resRL_0.05_contrsign_G <- lfcShrink(testdiff.genus.mucus,
#contrast=c("Treatment","RL_0.05","Control_0"),
# res=resRL_0.05_contr_G, type = "ashr")$padj<0.05
resRL_0.05_contrsignOTU_mucus_G <- rownames(genus.table.wnwn.mucus.treat)[
   resRL_0.05_contrsign_mucus_G==TRUE]

resRL_0.05_contrvol_mucus_G <- volcanoLisa(resRL_0.05_contr_mucus_G,
```

```
                                          taxlevel="genus",1,-1) +
    ggtitle("RL-0,05% \u2013 Control \u2013 Mucus") + theme(legend.box="vertical")
#plotting volcanoplot
resRL_0.05_contrvol_mucus_G
```

```
mylegend_volcano_RL_0.05_contr_G <- get_legend(resRL_0.05_contrvol_mucus_G)
resRL_0.05_contrvol_mucus_G <- papertheme(resRL_0.05_contrvol_mucus_G) +
    theme(legend.position="none",axis.line = element_line(color="black"))+
    scale_y_continuous(limits = c(0,30))
resRL_0.05_contrvol_mucus_G
```

```
## RL_0,5 vs control
resRL_0.5_contr_mucus_G <- results(testdiff.genus.mucus,
                                   contrast=c("Treatment","RL - 0.5","Control - 0"),
                                   test="Wald",alpha=0.05,independentFiltering=FALSE)
#shrinking LogFC-values so OTU's with low counts don't take over unnecessary attention.
resRL_0.5_contr_mucus_G <- lfcShrink(testdiff.genus.mucus,
                                     contrast=c("Treatment","RL - 0.5","Control - 0"),
                                     res=resRL_0.5_contr_mucus_G, type = "ashr")
#extract significantly different OTU's
resRL_0.5_contrsign_mucus_G <- resRL_0.5_contr_mucus_G$padj <0.05
#resRL_0.5_contrsign_G <- lfcShrink(testdiff.genus,
#contrast=c("Treatment","RL_0.5","Control_0"),
#res=resRL_0.5_contr_G,type = "ashr")$padj<0.05
resRL_0.5_contrsignOTU_mucus_G <- rownames(genus.table.wnwn.mucus.treat)[
  resRL_0.5_contrsign_mucus_G==TRUE]
resRL_0.5_contrsignOTU_mucus_G <- na.omit(resRL_0.5_contrsignOTU_mucus_G)

resRL_0.5_contrvol_mucus_G <- volcanoLisa(resRL_0.5_contr_mucus_G,
                                          taxlevel="genus",1,-1) +
    ggtitle("RL-0,5% \u2013 Control \u2013 Mucus") + theme(legend.box="vertical")
resRL_0.5_contrvol_mucus_G
```

```
  mylegend_volcano_RL_0.5_contr_mucus_G <- get_legend(resRL_0.5_contrvol_mucus_G)
resRL_0.5_contrvol_mucus_G <- papertheme(resRL_0.5_contrvol_mucus_G) +
    theme(legend.position="none",axis.line = element_line(color="black"))+
    scale_y_continuous(limits = c(0,30))
resRL_0.5_contrvol_mucus_G
```

```
source('/media/projects2/LisaM/Batch_Experiments/Illumina_Run3_Processing/Batch_Illumina_All/LM_volcano

#Turn volcanoplots into grobs to facilitate plotting.
volcanogrob_G1 <- ggplot_gtable(ggplot_build(resP80_0.05_contrvol_lumen_G))
volcanogrob_G2 <- ggplot_gtable(ggplot_build(resP80_0.5_contrvol_lumen_G))
volcanogrob_G3 <- ggplot_gtable(ggplot_build(resP80_0.05_contrvol_mucus_G))
volcanogrob_G4 <- ggplot_gtable(ggplot_build(resP80_0.5_contrvol_mucus_G))
volcanogrob_G5 <- ggplot_gtable(ggplot_build(resRL_0.05_contrvol_lumen_G))
volcanogrob_G6 <- ggplot_gtable(ggplot_build(resRL_0.5_contrvol_lumen_G))
volcanogrob_G7 <- ggplot_gtable(ggplot_build(resRL_0.05_contrvol_mucus_G))
volcanogrob_G8 <- ggplot_gtable(ggplot_build(resRL_0.5_contrvol_mucus_G))

dpi= 300
  # tiff("/media/projects2/LisaM/Batch_Experiments/Illumina_Run3_Processing/
#Batch_Illumina_All/Batch_Illumina_Deseq.tiff",width=16*dpi,height=18*dpi,
```

```
#res=dpi,compression="lzw")
  # ggdraw() + draw_plot(volcanogrobOTU1,0,0.8,0.5,0.33,0.17) +
#draw_plot(volcanogrobOTU2,0.33,0.8,0.33,0.17) +
#draw_plot(volcanogrobOTU3,0.66,0.8,0.33,0.17)
#draw_plot(mylegend_volcano_RL_0.5_contr,0.05,0,0.1,0.1)
  # dev.off()

library("grid")
library("gridExtra")

## P80

tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_Illumina_Deseq-Genus_
     width=16*dpi,height=12*dpi,res=dpi,compression="lzw")

Figure_Deseq_G <- grid.arrange(volcanogrob_G1,volcanogrob_G2,volcanogrob_G3,
                               volcanogrob_G4,ncol = 2, nrow = 2)
Figure_Deseq_G

titledeseq <- ggdraw() + draw_label("Normalised log transformed counts",angle=90,size=20)
plot_grid(titledeseq,
          plot_grid(Figure_Deseq_G,plot_grid(mylegend_volcano_P80_0.5_contr_lumen_G,ncol=1),
                    ncol=1,rel_heights=c(0.9,0.1)),ncol=2,rel_widths=c(0.05,0.95))
dev.off()

## RL

tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_Illumina_Deseq-Genus_
     width=16*dpi,height=12*dpi,res=dpi,compression="lzw")

Figure_Deseq_G <- grid.arrange(volcanogrob_G5,volcanogrob_G6,volcanogrob_G7,volcanogrob_G8,
                               ncol = 2, nrow = 2)
Figure_Deseq_G

titledeseq <- ggdraw() + draw_label("Normalised log transformed counts",angle=90,size=20)
plot_grid(titledeseq,plot_grid(Figure_Deseq_G,plot_grid(
  mylegend_volcano_P80_0.5_contr_lumen_G,ncol=1),ncol=1,rel_heights=c(0.9,0.1)),
  ncol=2,rel_widths=c(0.05,0.95))
dev.off()
```

14.3 BARPLOTS

14.3.1 LUMEN

```
# Concatenate all pairwise comparisons
genusnames.P80.RL.lumen <- unique(c(resP80_0.05_contrsignOTU_lumen_G,
                                    resP80_0.5_contrsignOTU_lumen_G,
                                    resRL_0.05_contrsignOTU_lumen_G,
                                    resRL_0.5_contrsignOTU_lumen_G))
#names of all up- or downregulated genusses's
genusnames.P80.RL.lumen <- na.omit(genusnames.P80.RL.lumen)
```

```
phylumname_P80.RL <- mapvalues(genusnames.P80.RL.lumen,
                               from=as.character(taxonomy.np.ns$Genus),
                               to=as.character(taxonomy.np.ns$Phylum))
# look up OTU-names in the rownames of taxonomy and extract corresponding phylumname
combinedgenus_phylum_P80.RL <- data.frame(genus=genusnames.P80.RL.lumen,
                                          phylum=phylumname_P80.RL)

#create dataframe containing all counts for all sign up-or downregulated
#genera's for all samples + metadata + significance letters

savegenus <- data.frame()

Pvalues <- data.frame()

for (i in genusnames.P80.RL.lumen){
pvalueP80_0.05_contr_lumen_G <- resP80_0.05_contr_lumen_G[i,]$padj
names(pvalueP80_0.05_contr_lumen_G) <- c("P80_0.05-contr")
pvalueP80_0.5_contr_lumen_G <- resP80_0.5_contr_lumen_G[i,]$padj
names(pvalueP80_0.5_contr_lumen_G) <- c("P80_0.5-contr")


pvalueRL_0.05_contr_lumen_G <- resRL_0.05_contr_lumen_G[i,]$padj
names(pvalueRL_0.05_contr_lumen_G) <- c("RL_0.05-contr")
pvalueRL_0.5_contr_lumen_G <- resRL_0.5_contr_lumen_G[i,]$padj
names(pvalueRL_0.5_contr_lumen_G) <- c("RL_0.5-contr")
pvaluetotal <- c(pvalueP80_0.05_contr_lumen_G,pvalueP80_0.5_contr_lumen_G,
                 pvalueRL_0.05_contr_lumen_G, pvalueRL_0.5_contr_lumen_G)

if(any(is.na(pvaluetotal))==FALSE){letters <- multcompLetters(pvaluetotal)$"Letters"}
else{letters <- rep("NA",4)}

#extracts counts for the selected genera for all samples
diffgenus.lumen <- plotCounts(testdiff.genus.lumen,gene=i,intgroup="Treatment",
                              returnData=TRUE)
diffgenus.lumen <- cbind(diffgenus.lumen,sa=metadata.lumen.treat)
diffgenus.lumen <- mutate(diffgenus.lumen,
                          phylum=rep(combinedgenus_phylum_P80.RL[
                            combinedgenus_phylum_P80.RL$genus==i,]$phylum,
                            nrow(diffgenus.lumen)),
                          genus=rep(combinedgenus_phylum_P80.RL[
                            combinedgenus_phylum_P80.RL$genus==i,]$genus,
                            nrow(diffgenus.lumen)),genus=i)

# diffgenus.mucus <- plotCounts(testdiff.genus.mucus,gene=i,intgroup="Treatment",
# returnData=TRUE) #extracts counts for the selected genera for all samples
# diffgenus.mucus <- cbind(diffgenus.mucus,sa=metadata.mucus.treat)
# diffgenus.mucus <- mutate(diffgenus.mucus,phylum=rep(combinedgenus_phylum_P80[
#combinedgenus_phylum_P80$genus==i,]$phylum,nrow(diffgenus.mucus)),
# genus=rep(combinedgenus_phylum_P80[combinedgenus_phylum_P80$genus==i,]$genus,
# nrow(diffgenus.mucus)),genus=i)

diffgenus.lumen$pval<- mapvalues(diffgenus.lumen$Treatment,
                                 from=unique(diffgenus.lumen$Treatment),
```

```
                                  to=c(1,pvaluetotal[1],pvaluetotal[2],
                                       pvaluetotal[3],pvaluetotal[4]))
diffgenus.lumen$labels<- mapvalues(diffgenus.lumen$Treatment,
                                   from=unique(diffgenus.lumen$Treatment),
                                   to=c("Z",letters[1],letters[2],letters[3],
                                        letters[4]))
savegenus <- rbind(savegenus,diffgenus.lumen)
Pvalues <- rbind(Pvalues,pvaluetotal)
}
savegenus

X.order.sign <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05","RL - 0.5")
# EM_Colors.sign <- c("#cecb99","#5d6295B3", "#c9e5ff", "#953d6aB3")

colnames(Pvalues)[1:4] <- X.order.sign[2:5]
Pvalues$Genus <- genusnames.P80.RL.lumen
write.xlsx(Pvalues,"SHIME_Deseq_pvalues_RL.P80_Lumen.xlsx")

# Prepare data for plotting
savegenus <- ddply(savegenus,.(savegenus$Treatment,genus),mutate,
                   Q1=quantile(log(count),1/4), Q3=quantile(log(count),3/4),
                   IQR=Q3-Q1,upper.limit=Q3+1.5*IQR,lower.limit=Q1-1.5*IQR)
#19 variables
savegenus <- savegenus[mixedorder(savegenus$genus),]
savegenusX <- savegenus

savegenusX <- savegenusX[,-1]
savegenusX <- savegenusX[,-3]
savegenusX <- savegenusX[,-3]
savegenus <- savegenusX
colnames(savegenus)[1:9] <- c("count","Treatment","Emulsifier","EM-Conc","SHIME",
                              "Timepoint","TimeCode","Location","Donor")

savegenus$pval <- as.numeric(levels(savegenus$pval)[savegenus$pval])
savegenus[is.na(savegenus)]<-1
savegenus$label <- gtools::stars.pval(savegenus$pval)
unique(savegenus$genus)

#  [1] "Agathobacter"                    "Akkermansia"                      "Alistipes"
# "Anaeroglobus"                     "Anaerostipes"
#  [6] "Bacteroidales_unclassified"      "Bacteroides"
#"Bifidobacteriaceae_unclassified" "Bifidobacterium"                  "Bilophila"
# [11] "Blautia"                         "Butyricicoccaceae_unclassified"
#"Butyricicoccus"                   "Candidatus_Soleaferrea"           "Centipeda"
# [16] "Clostridia_unclassified"         "Clostridiaceae_unclassified"
#"Colidextribacter"                 "Collinsella"                      "Coprococcus"
# [21] "Dorea"                           "Eisenbergiella"
#"Enterobacteriaceae_unclassified" "Erysipelatoclostridium"           "Faecalibacterium"
# [26] "Fusicatenibacter"                "Gastranaerophilales_ge"
#"Holdemanella"                     "Holdemania"                       "Hungatella"
# [31] "Lachnoclostridium"               "Lachnospira"
#"Lachnospiraceae_ge"               "Lachnospiraceae_ND3007_group"
#"Lachnospiraceae_unclassified"
```

```
# [36] "Lactobacillales_unclassified"    "Lactobacillus"
#"Megasphaera"                        "Micrococcaceae_unclassified"     "Negativibacillus"
# [41] "Oscillibacter"                       "Oscillospira"
#"Oscillospirales_unclassified"     "Parabacteroides"                 "Paraprevotella"
# [46] "Prevotella"                          "Prevotellaceae_unclassified"
#"Ruminococcaceae_unclassified"     "Subdoligranulum"                 "Sutterella"
# [51] "Tyzzerella"                          "UBA1819"
#"UCG-003"                           "UCG-002"


# Order data in subpanels of multipanel graph - LEAVE OUT genus 15,21,30,35,39,41,
#42,44,62,69,84,86,87

savegenus1<-savegenus[savegenus$genus%in%c("Agathobacter","Akkermansia","Alistipes",
                                           "Anaeroglobus","Anaerostipes",
                                           "Bacteroidales_unclassified"),]
savegenus1<- savegenus1[mixedorder(savegenus1$genus),]
savegenus1$otu <- reorder(factor(savegenus1$genus),sort=mixedsort)

savegenus2<-savegenus[savegenus$genus%in%c("Bacteroides",
                                           "Bifidobacteriaceae_unclassified",
                                           "Bifidobacterium","Bilophila","Blautia",
                                           "Butyricicoccaceae_unclassified"),]
savegenus2<- savegenus2[mixedorder(savegenus2$genus),]
savegenus2$otu <- reorder(factor(savegenus2$genus),sort=mixedsort)

savegenus3<-savegenus[savegenus$genus%in%c("Butyricicoccus","Candidatus_Soleaferrea",
                                           "Centipeda","Clostridia_unclassified",
                                           "Clostridiaceae_unclassified",
                                           "Colidextribacter"),]
savegenus3<- savegenus3[mixedorder(savegenus3$genus),]
savegenus3$otu <- reorder(factor(savegenus3$genus),sort=mixedsort)

savegenus4<-savegenus[savegenus$genus%in%c("Collinsella","Coprococcus","Dorea",
                                           "Eisenbergiella",
                                           "Enterobacteriaceae_unclassified",
                                           "Erysipelatoclostridium"),]
savegenus4<- savegenus4[mixedorder(savegenus4$genus),]
savegenus4$otu <- reorder(factor(savegenus4$genus),sort=mixedsort)

savegenus5<-savegenus[savegenus$genus%in%c("Faecalibacterium","Fusicatenibacter",
                                           "Gastranaerophilales_ge","Holdemanella",
                                           "Holdemania","Hungatella"),]
savegenus5<- savegenus5[mixedorder(savegenus5$genus),]
savegenus5$otu <- reorder(factor(savegenus5$genus),sort=mixedsort)

savegenus6<-savegenus[savegenus$genus%in%c("Lachnoclostridium","Lachnospira",
                                           "Lachnospiraceae_ge",
                                           "Lachnospiraceae_ND3007_group",
                                           "Lachnospiraceae_unclassified",
                                           "Lactobacillales_unclassified"),]
savegenus6<- savegenus6[mixedorder(savegenus6$genus),]
savegenus6$otu <- reorder(factor(savegenus6$genus),sort=mixedsort)
```

```
savegenus7<-savegenus[savegenus$genus%in%c("Lactobacillus","Megasphaera",
                                           "Micrococcaceae_unclassified",
                                           "Negativibacillus","Oscillibacter",
                                           "Oscillospira"),]
savegenus7<- savegenus7[mixedorder(savegenus7$genus),]
savegenus7$otu <- reorder(factor(savegenus7$genus),sort=mixedsort)

savegenus8<-savegenus[savegenus$genus%in%c("Oscillospirales_unclassified",
                                           "Parabacteroides","Paraprevotella",
                                           "Prevotella",
                                           "Prevotellaceae_unclassified",
                                           "Ruminococcaceae_unclassified"),]
savegenus8<- savegenus8[mixedorder(savegenus8$genus),]
savegenus8$otu <- reorder(factor(savegenus8$genus),sort=mixedsort)

savegenus9<-savegenus[savegenus$genus%in%c("Subdoligranulum","Sutterella","Tyzzerella",
                                           "UBA1819","UCG-003","UCG-002"),]
savegenus9<- savegenus9[mixedorder(savegenus9$genus),]
savegenus9$otu <- reorder(factor(savegenus9$genus),sort=mixedsort)

#magma colors
# EM_Colors <- c("#837239B3", "#AB337CFF","#cecb99","#681D81FF", "#FEB37BFF", "#D6456CFF")
#                 CMC           Control    tween80    RL          SL      SoyLec

X.order.sign <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05","RL - 0.5")
# EM_Colors.sign <- c("#cecb99","#5d6295B3", "#c9e5ff", "#953d6aB3")

#original colors
EM_Colors <- c("#999999","#A660CC", "#4EAC57", "#E1A13F", "#A660CC", "#21AADE")

#CD5C5C = red
#4EAC57 = green
#21AADE = blue
#A660CC = purple
#E1A13F = orange
#999999 = grey

savegenus1$genus <- mapvalues(savegenus1$genus, from = "Bacteroidales_unclassified",
                              to = "Bacteroidales unclassified")
savegenus1$otu <- mapvalues(savegenus1$otu, from = "Bacteroidales_unclassified",
                            to = "Bacteroidales unclassified")

plot_G1 <- ggplot(savegenus1,aes(x=Treatment,y=count,fill=Emulsifier))+
  geom_boxplot()+
  #geom_point(data=saveotu1[log(saveotu1$count)>saveotu1$upper.limit|
  #log(saveotu1$count)<saveotu1$lower.limit,],aes(x=Treatment,y=count,color=Treatment))+
  scale_x_discrete(limits=X.order.sign)

plot_G1 <- plot_G1 + scale_y_continuous(labels=scientific,trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free', labeller = label_wrap_gen(20))
```

```
plot_G1 <- papertheme(plot_G1)+
  #theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=NC_PD_colors,name=NULL)+
  #theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"),
        axis.title.x = element_blank())+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),size=3, color = "black")
plot_G1
```

```
mylegenddeseq <- get_legend(plot_G1)

# deseqsaveotu1 <-plot_G1+ theme(legend.position = "none",
#axis.line = element_line(color="black"))
# deseqsaveotu1strips <- ggplot_gtable(ggplot_build(deseqsaveotu1))
# stripr <- which(grepl('strip-t',deseqsaveotu1strips$layout$name)) #checks if "strip-t" can be found i
# fills <- as.character(saveotufillphylum1$fill)
# k <- 1
# for (i in stripr) {
#   j <- which(grepl('rect', deseqsaveotu1strips$grobs[[i]]$grobs[[1]]$childrenOrder))
#   deseqsaveotu1strips$grobs[[i]]$grobs[[1]]$children[[j]]$gp$fill <- fills[k]
#   k <- k+1
# }
# grid.draw(deseqsaveotu1strips)

savegenus2$genus <- mapvalues(savegenus2$genus, from = "Bifidobacteriaceae_unclassified",
                              to = "Bifidobacteriaceae unclassified")
savegenus2$otu <- mapvalues(savegenus2$otu, from = "Bifidobacteriaceae_unclassified",
                            to = "Bifidobacteriaceae unclassified")

savegenus2$genus <- mapvalues(savegenus2$genus, from = "Butyricicoccaceae_unclassified",
                              to = "Butyricicoccaceae unclassified")
savegenus2$otu <- mapvalues(savegenus2$otu, from = "Butyricicoccaceae_unclassified",
                            to = "Butyricicoccaceae unclassified")

plot_G2 <- ggplot(savegenus2,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G2 <- plot_G2 + scale_y_continuous(labels=scientific,trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+

  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free', labeller = label_wrap_gen(20))

plot_G2 <- papertheme(plot_G2)+theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),
```

```
        axis.title.y = element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10), color = "black")
plot_G2
```

```
# deseqsaveotu2 <-plot2+ theme(legend.position = "none",
#axis.line = element_line(color="black"))
# deseqsaveotu2strips<- ggplot_gtable(ggplot_build(deseqsaveotu2))
# stripr <- which(grepl('strip-t',deseqsaveotu2strips$layout$name))
# fills <- as.character(saveotufillphylum2$fill)
# k <- 1
# for (i in stripr) {
#   j <- which(grepl('rect', deseqsaveotu2strips$grobs[[i]]$grobs[[1]]$childrenOrder))
#   deseqsaveotu2strips$grobs[[i]]$grobs[[1]]$children[[j]]$gp$fill <- fills[k]
#   k <- k+1
# }
# grid.draw(deseqsaveotu2strips)

savegenus3$genus <- mapvalues(savegenus3$genus, from = "Clostridia_unclassified",
                              to = "Clostridia unclassified")
savegenus3$otu <- mapvalues(savegenus3$otu, from = "Clostridia_unclassified",
                            to = "Clostridia unclassified")

savegenus3$genus <- mapvalues(savegenus3$genus, from = "Clostridiaceae_unclassified",
                              to = "Clostridiaceae unclassified")
savegenus3$otu <- mapvalues(savegenus3$otu, from = "Clostridiaceae_unclassified",
                            to = "Clostridiaceae unclassified")

savegenus3$genus <- mapvalues(savegenus3$genus, from = "Candidatus_Soleaferrea",
                              to = "Candidatus Soleaferre")
savegenus3$otu <- mapvalues(savegenus3$otu, from = "Candidatus_Soleaferrea",
                            to = "Candidatus Soleaferrea")

plot_G3 <- ggplot(savegenus3,aes(x=Treatment,y=count,fill=Emulsifier))+
  geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G3 <- plot_G3 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                             labeller = labeller(otu = label_wrap_gen(20)))

plot_G3 <- papertheme(plot_G3)+theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y =  element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10), color = "black")
plot_G3
```

```
savegenus4$genus <- mapvalues(savegenus4$genus, from = "Enterobacteriaceae_unclassified",
                              to = "Enterobacteriaceae unclassified")
savegenus4$otu <- mapvalues(savegenus4$otu, from = "Enterobacteriaceae_unclassified",
                            to = "Enterobacteriaceae unclassified")

plot_G4 <- ggplot(savegenus4,aes(x=Treatment,y=count,fill=Emulsifier)) + geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G4 <- plot_G4 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G4 <- papertheme(plot_G4) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none") +
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),

plot_G4
```

```
savegenus5$genus <- mapvalues(savegenus5$genus, from = "Gastranaerophilales_ge",
                              to = "Gastranaerophilales ge")
savegenus5$otu <- mapvalues(savegenus5$otu, from = "Gastranaerophilales_ge",
                            to = "Gastranaerophilales ge")

plot_G5 <- ggplot(savegenus5,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G5 <- plot_G5 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G5 <- papertheme(plot_G5) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G5
```

```
savegenus6$genus <- mapvalues(savegenus6$genus, from = "Lachnospiraceae_ge",
                              to = "Lachnospiraceae ge")
```

```
savegenus6$otu <- mapvalues(savegenus6$otu, from = "Lachnospiraceae_ge",
                            to = "Lachnospiraceae ge")

savegenus6$genus <- mapvalues(savegenus6$genus, from = "Lachnospiraceae_ND3007_group",
                              to = "Lachnospiraceae ND3007 group")
savegenus6$otu <- mapvalues(savegenus6$otu, from = "Lachnospiraceae_ND3007_group",
                            to = "Lachnospiraceae ND3007 group")

savegenus6$genus <- mapvalues(savegenus6$genus, from = "Lachnospiraceae_unclassified",
                              to = "Lachnospiraceae unclassified")
savegenus6$otu <- mapvalues(savegenus6$otu, from = "Lachnospiraceae_unclassified",
                            to = "Lachnospiraceae unclassified")

savegenus6$genus <- mapvalues(savegenus6$genus, from = "Lactobacillales_unclassified",
                              to = "Lactobacillales unclassified")
savegenus6$otu <- mapvalues(savegenus6$otu, from = "Lactobacillales_unclassified",
                            to = "Lactobacillales unclassified")

plot_G6 <- ggplot(savegenus6,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|log(saveotu2$count)<saveotu2$lower
plot_G6 <- plot_G6 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G6 <- papertheme(plot_G6) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G6
```

```
savegenus7$genus <- mapvalues(savegenus7$genus, from = "Micrococcaceae_unclassified",
                              to = "Micrococcaceae unclassified")
savegenus7$otu <- mapvalues(savegenus7$otu, from = "Micrococcaceae_unclassified",
                            to = "Micrococcaceae unclassified")

plot_G7 <- ggplot(savegenus7,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G7 <- plot_G7 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G7 <- papertheme(plot_G7) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
```

```
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G7
```

```
savegenus8$genus <- mapvalues(savegenus8$genus, from = "Oscillospirales_unclassified",
                              to = "Oscillospirales unclassified")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Oscillospirales_unclassified",
                            to = "Oscillospirales unclassified")

savegenus8$genus <- mapvalues(savegenus8$genus, from = "Prevotellaceae_unclassified",
                              to = "Prevotellaceae unclassified")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Prevotellaceae_unclassified",
                            to = "Prevotellaceae unclassified")

savegenus8$genus <- mapvalues(savegenus8$genus, from = "Ruminococcaceae_unclassified",
                              to = "Ruminococcaceae unclassified")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Ruminococcaceae_unclassified",
                            to = "Ruminococcaceae unclassified")

plot_G8 <- ggplot(savegenus8,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G8 <- plot_G8 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G8 <- papertheme(plot_G8) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G8
```

```
savegenus9$genus <- mapvalues(savegenus9$genus, from = "Prevotellaceae_unclassified",
                              to = "Prevotellaceae unclassified")
savegenus9$otu <- mapvalues(savegenus9$otu, from = "Prevotellaceae_unclassified",
                            to = "Prevotellaceae unclassified")

plot_G9 <- ggplot(savegenus9,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G9 <- plot_G9 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),trans='log10',
```

```
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G9 <- papertheme(plot_G9) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "bottom")+
theme(strip.text.x=element_text(size=14),
      axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G9
```

```
plot_X <- ggplot(savegenus6,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot() +
  scale_x_discrete(limits=X.order.sign, drop=TRUE)+
  scale_fill_manual(breaks=c("Control","RL","P80"),values=c("#999999","#E1A13F","#4EAC57",
                                                            "#A660CC","#21AADE"),
                    name="Emulsifier")+ guides(fill = guide_legend(reverse = TRUE))+
  theme(legend.text = element_text(size=12,family='sans',angle=0),
        legend.title = element_text(size=14,family='sans',angle=0))
plot_X
```

```
mylegenddeseq <- get_legend(plot_X)

#original colors
EM_Colors <- c("#CD5C5C", "#999999", "#4EAC57", "#E1A13F", "#A660CC", "#21AADE")
#                 CMC           Control    tween80    RL          SL       SoyLec

#EM_Colors <- c("#837239B3", "#557d7aB3","#cecb99","#5d6295B3", "#c9e5ff", "#953d6aB3")
                  #brown     #green    #yellow     #purple    #light blue   #red
                  #CMC        #Control    #P80        #RL         #SL       SoyLec

# Multi plot of bargraphs
titledeseq <- ggdraw() + draw_label("Normalised log transformed counts",angle=90,size=20)

deseqgenusotuselect1 <- ggdraw() + draw_plot(plot_G1,0,0.9,1,0.10) +
  draw_plot(plot_G2,0,0.8,1,0.10) + draw_plot(plot_G3,0,0.7,1,0.10) +
  draw_plot(plot_G4,0,0.6,1,0.10) + draw_plot(plot_G5,0,0.5,1,0.10) +
  draw_plot(plot_G6,0,0.4,1,0.10) + draw_plot(plot_G7,0,0.3,1,0.10) +
  draw_plot(plot_G8,0,0.20,1,0.10) + draw_plot(plot_G9,0,0,1,0.20)

tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_DESeq_Genera_Barplot_
     width=13*dpi,height=16.25*dpi,res=dpi,compression="lzw")
plot_grid(titledeseq,deseqgenusotuselect1,ncol=2,rel_widths=c(0.02,0.98))
dev.off()


# Get Log2FoldChange
Log2FoldChange <- data.frame()

for (i in genusnames.P80.RL.lumen){
```

```
LFCP80_0.05_contr_lumen_G <- resP80_0.05_contr_lumen_G[i,]$log2FoldChange
names(LFCP80_0.05_contr_lumen_G) <- c("P80_0.05-contr")
LFCP80_0.5_contr_lumen_G <- resP80_0.5_contr_lumen_G[i,]$log2FoldChange
names(LFCP80_0.5_contr_lumen_G) <- c("P80_0.5-contr")

LFCRL_0.05_contr_lumen_G <- resRL_0.05_contr_lumen_G[i,]$log2FoldChange
names(LFCRL_0.05_contr_lumen_G) <- c("RL_0.05-contr")
LFCRL_0.5_contr_lumen_G <- resRL_0.5_contr_lumen_G[i,]$log2FoldChange
names(LFCRL_0.5_contr_lumen_G) <- c("RL_0.5-contr")

L2FCtotal <- c(LFCP80_0.05_contr_lumen_G, LFCP80_0.5_contr_lumen_G,
               LFCRL_0.05_contr_lumen_G, LFCRL_0.5_contr_lumen_G)

Log2FoldChange <- rbind(Log2FoldChange,L2FCtotal)
}

colnames(Log2FoldChange)[1:4] <- X.order.sign[2:5]
Log2FoldChange$Genus <- genusnames.P80.RL.lumen
write.xlsx(Log2FoldChange,"Deseq_Log2FoldChange_RL_P80_Lumen.xlsx")
```

14.3.2 MUCUS

```
# Concatenate all pairwise comparisons
genusnames.P80.RL.mucus <- unique(c(resP80_0.05_contrsignOTU_mucus_G,
                                    resP80_0.5_contrsignOTU_mucus_G,
                                    resRL_0.05_contrsignOTU_mucus_G,
                                    resRL_0.5_contrsignOTU_mucus_G))
#names of all up- or downregulated genusses's
genusnames.P80.RL.mucus <- na.omit(genusnames.P80.RL.mucus)

# look up OTU-names in the rownames of taxonomy and extract corresponding phylumname
phylumname_RL <- mapvalues(genusnames.P80.RL.mucus,from=as.character(taxonomy.np.ns$Genus),
                           to=as.character(taxonomy.np.ns$Phylum))
combinedgenus_phylum_RL <- data.frame(genus=genusnames.P80.RL.mucus,phylum=phylumname_RL)

#create dataframe containing all counts for all sign up-or downregulated genera's for all
#samples + metadata + significance letters

savegenus <- data.frame()

Pvalues <- data.frame()

for (i in genusnames.P80.RL.mucus){
 pvalueP80_0.05_contr_mucus_G <- resP80_0.05_contr_mucus_G[i,]$padj
 names(pvalueP80_0.05_contr_mucus_G) <- c("P80_0.05-contr")
 pvalueP80_0.5_contr_mucus_G <- resP80_0.5_contr_mucus_G[i,]$padj
 names(pvalueP80_0.5_contr_mucus_G) <- c("P80_0.5-contr")

 pvalueRL_0.05_contr_mucus_G <- resRL_0.05_contr_mucus_G[i,]$padj
 names(pvalueRL_0.05_contr_mucus_G) <- c("RL_0.05-contr")
 pvalueRL_0.5_contr_mucus_G <- resRL_0.5_contr_mucus_G[i,]$padj
 names(pvalueRL_0.5_contr_mucus_G) <- c("RL_0.5-contr")
```

```
 pvaluetotal <- c(pvalueP80_0.05_contr_mucus_G,pvalueP80_0.5_contr_mucus_G,
                  pvalueRL_0.05_contr_mucus_G, pvalueRL_0.5_contr_mucus_G)

 if(any(is.na(pvaluetotal))==FALSE){letters <- multcompLetters(pvaluetotal)$"Letters"}
 else{letters <- rep("NA",4)}


diffgenus.mucus <- plotCounts(testdiff.genus.mucus,gene=i,intgroup="Treatment",
                              returnData=TRUE)
#extracts counts for the selected genera for all samples
diffgenus.mucus <- cbind(diffgenus.mucus,sa=metadata.mucus.treat)
diffgenus.mucus <- mutate(diffgenus.mucus,
                  phylum=rep(combinedgenus_phylum_RL[combinedgenus_phylum_RL$genus==i,]$phylum,
                                  nrow(diffgenus.mucus)),
                  genus=rep(combinedgenus_phylum_RL[combinedgenus_phylum_RL$genus==i,]$genus,
                                 nrow(diffgenus.mucus)),genus=i)

diffgenus.mucus$pval<- mapvalues(diffgenus.mucus$Treatment,
                                 from=unique(diffgenus.mucus$Treatment),
                                 to=c(1,pvaluetotal[1],pvaluetotal[2],pvaluetotal[3],
                                      pvaluetotal[4]))

diffgenus.mucus$labels<- mapvalues(diffgenus.mucus$Treatment,
                                   from=unique(diffgenus.mucus$Treatment),
                                   to=c("Z",letters[1],letters[2],letters[3],letters[4]))
savegenus <- rbind(savegenus,diffgenus.mucus)
Pvalues <- rbind(Pvalues,pvaluetotal)
}
savegenus

colnames(Pvalues)[1:4] <- X.order.sign[2:5]
Pvalues$Genus <- genusnames.P80.RL.mucus
write.xlsx(Pvalues,"SHIME_Deseq_pvalues_RL.P80_Mucus.xlsx")

#saveotu$NC_PD <-factor(saveotu$NC_PD,levels=c("NRProximal","NRDistal","NLProximal",
#"NLDistal","NDProximal","NDDistal"))
#saveotu$NC_PD <- mapvalues(saveotu$NC_PD,from=c("NRProximal","NRDistal","NLProximal",
#"NLDistal","NDProximal","NDDistal"),to=c("NR P","NR D","NL P","NL D","ND P","ND D"))

# Prepare data for plotting
savegenus <- ddply(savegenus,.(savegenus$Treatment,genus),mutate, Q1=quantile(log(count),1/4), Q3=quant
# 19 variables
savegenus <- savegenus[mixedorder(savegenus$genus),]
savegenusX <- savegenus

savegenusX <- savegenusX[,-1]
savegenusX <- savegenusX[,-3]
savegenusX <- savegenusX[,-3]
savegenus <- savegenusX
colnames(savegenus)[1:9] <- c("count","Treatment","Emulsifier","EM-Conc","SHIME",
                              "Timepoint","TimeCode","Location","Donor")

savegenus$pval <- as.numeric(levels(savegenus$pval)[savegenus$pval])
```

```
savegenus[is.na(savegenus)]<-1
savegenus$label <- gtools::stars.pval(savegenus$pval)
unique(savegenus$genus)
#  [1] "Acetanaerobacterium"              "Agathobacter"                      "Akkermansia"
#"Alistipes"                           "Anaeroglobus"
#  [6] "Anaerostipes"                   "Barnesiella"                       "Bifidobacterium"
#"Bilophila"                           "Blautia"
# [11] "CAG-56"                         "Candidatus_Soleaferrea"            "Centipeda"
#"Cloacibacillus"                      "Clostridia_unclassified"
# [16] "Colidextribacter"               "Collinsella"                       "Coprococcus"
#"Desulfovibrio"                       "Desulfovibrionaceae_unclassified"
# [21] "Dorea"                          "Eisenbergiella"            "Enterobacterales_unclassified"
#"Enterobacteriaceae_unclassified"     "Erysipelotrichaceae_ge"
# [26] "Faecalibacterium"               "Flavonifractor"              "Fusicatenibacter"
#"Gastranaerophilales_ge"              "Holdemanella"
# [31] "Holdemania"                     "Hungatella"                        "Intestinimonas"
#"Lachnospira"                         "Lachnospiraceae_ge"
# [36] "Lachnospiraceae_ND3007_group" "Lachnospiraceae_unclassified""Lactobacillales_unclassified"
#"Lactobacillus"                       "Marvinbryantia"
# [41] "Megamonas"                      "Megasphaera"                       "Negativibacillus"
#"Odoribacter"                         "Oscillibacter"
# [46] "Oscillospira"    "Oscillospiraceae_unclassified"    "Oscillospirales_unclassified"
#"Parabacteroides"                     "Paraprevotella"
# [51] "Phocea"         "Prevotella"  "Prevotellaceae_unclassified"
#"Pseudomonas"                         "Ruminococcaceae_unclassified"
# [56] "Selenomonadaceae_unclassified"    "Subdoligranulum"                   "Tyzzerella"
# "UBA1819"                             "UCG-003"
# [61] "UCG-002"                        "uncultured_ge"

# Order data in subpanels of multipanel graph - LEAVE OUT genus 15,21,30,35,39,41,42,
#44,62,69,84,86,87

savegenus1<-savegenus[savegenus$genus%in%c("Acetanaerobacterium","Agathobacter",
                                           "Akkermansia","Alistipes","Anaeroglobus"),]
savegenus1<- savegenus1[mixedorder(savegenus1$genus),]
savegenus1$otu <- reorder(factor(savegenus1$genus),sort=mixedsort)

savegenus2<-savegenus[savegenus$genus%in%c("Anaerostipes","Barnesiella","Bifidobacterium",
                                           "Bilophila","Blautia"),]
savegenus2<- savegenus2[mixedorder(savegenus2$genus),]
savegenus2$otu <- reorder(factor(savegenus2$genus),sort=mixedsort)

savegenus3<-savegenus[savegenus$genus%in%c("CAG-56","Candidatus_Soleaferrea","Centipeda",
                                           "Cloacibacillus","Clostridia_unclassified"),]
savegenus3<- savegenus3[mixedorder(savegenus3$genus),]
savegenus3$otu <- reorder(factor(savegenus3$genus),sort=mixedsort)

savegenus4<-savegenus[savegenus$genus%in%c("Colidextribacter","Collinsella","Coprococcus",
                                           "Desulfovibrio",
                                           "Desulfovibrionaceae_unclassified"),]
savegenus4<- savegenus4[mixedorder(savegenus4$genus),]
savegenus4$otu <- reorder(factor(savegenus4$genus),sort=mixedsort)
```

```
savegenus5<-savegenus[savegenus$genus%in%c("Dorea","Eisenbergiella",
                                           "Enterobacterales_unclassified",
                                           "Enterobacteriaceae_unclassified",
                                           "Erysipelotrichaceae_ge"),]
savegenus5<- savegenus5[mixedorder(savegenus5$genus),]
savegenus5$otu <- reorder(factor(savegenus5$genus),sort=mixedsort)

savegenus6<-savegenus[savegenus$genus%in%c("Faecalibacterium","Flavonifractor",
                                           "Fusicatenibacter","Gastranaerophilales_ge",
                                           "Holdemanella"),]
savegenus6<- savegenus6[mixedorder(savegenus6$genus),]
savegenus6$otu <- reorder(factor(savegenus6$genus),sort=mixedsort)

savegenus7<-savegenus[savegenus$genus%in%c("Holdemania","Hungatella","Intestinimonas",
                                           "Lachnospira","Lachnospiraceae_ge"),]
savegenus7<- savegenus7[mixedorder(savegenus7$genus),]
savegenus7$otu <- reorder(factor(savegenus7$genus),sort=mixedsort)

savegenus8<-savegenus[savegenus$genus%in%c("Lachnospiraceae_ND3007_group",
                                           "Lachnospiraceae_unclassified",
                                           "Lactobacillales_unclassified","Lactobacillus",
                                           "Marvinbryantia"),]
savegenus8<- savegenus8[mixedorder(savegenus8$genus),]
savegenus8$otu <- reorder(factor(savegenus8$genus),sort=mixedsort)

savegenus9<-savegenus[savegenus$genus%in%c( "Megamonas","Megasphaera","Negativibacillus",
                                           "Odoribacter","Oscillibacter" ),]
savegenus9<- savegenus9[mixedorder(savegenus9$genus),]
savegenus9$otu <- reorder(factor(savegenus9$genus),sort=mixedsort)

savegenus10<-savegenus[savegenus$genus%in%c("Oscillospira",
                                           "Oscillospiraceae_unclassified",
                                            "Oscillospirales_unclassified",
                                            "Parabacteroides","Paraprevotella"),]
savegenus10<- savegenus10[mixedorder(savegenus10$genus),]
savegenus10$otu <- reorder(factor(savegenus10$genus),sort=mixedsort)

savegenus11<-savegenus[savegenus$genus%in%c("Phocea","Prevotella",
                                           "Prevotellaceae_unclassified","Pseudomonas",
                                            "Ruminococcaceae_unclassified"),]
savegenus11<- savegenus11[mixedorder(savegenus11$genus),]
savegenus11$otu <- reorder(factor(savegenus11$genus),sort=mixedsort)

savegenus12<-savegenus[savegenus$genus%in%c("Selenomonadaceae_unclassified",
                                            "Subdoligranulum","Tyzzerella","UBA1819",
                                            "UCG-003"),]
savegenus12<- savegenus12[mixedorder(savegenus12$genus),]
savegenus12$otu <- reorder(factor(savegenus12$genus),sort=mixedsort)

savegenus13<-savegenus[savegenus$genus%in%c("UCG-002","uncultured_ge"),]
savegenus13<- savegenus13[mixedorder(savegenus13$genus),]
savegenus13$otu <- reorder(factor(savegenus13$genus),sort=mixedsort)
```

```
# Color strips by phylum level classification
# saveotufillphylum <- rbind(saveotu1,saveotu2,saveotu3,saveotu4,saveotu5,saveotu6,
#saveotu7,saveotu8)
# saveotufillphylum$fill <- mapvalues(saveotufillphylum$phylum,
#from=levels(saveotufillphylum$phylum),to=c("#837239B3","#557d7aB3","#cecb99","#5d6295B3",
#"#c9e5ff","#953d6aB3","#efdcdd")) #assigns a color to each phylum
# saveotufillphylum <-saveotufillphylum[!duplicated(saveotufillphylum$otu),]
#removes duplicate OTU's?
#
# plot <- ggplot(saveotufillphylum,aes(x=NC_PD,y=count,
#fill=saveotufillphylum$phylum))+ geom_boxplot() +
# scale_fill_manual(breaks=levels(saveotufillphylum$phylum),
#values=c(Actinobacteria="#837239B3",Bacteria_unclassified="#557d7aB3",
# Bacteroidetes="#cecb99",Firmicutes="#5d6295B3",Fusobacteria="#c9e5ff",
# Proteobacteria="#953d6aB3",Verrucomicrobia="#efdcdd"),name="Phylum")+
# theme(legend.position="bottom",legend.text=element_text(face='italic'))+
# guides(fill=guide_legend(nrow=3))
# striplegend <- get_legend(plot)
# plot
# saveotufillphylum1 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu1$otu,]
# saveotufillphylum2 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu2$otu,]
# saveotufillphylum3 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu3$otu,]
# saveotufillphylum4 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu4$otu,]
# saveotufillphylum5 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu5$otu,]
# saveotufillphylum6 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu6$otu,]
# saveotufillphylum7 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu7$otu,]
# saveotufillphylum8 <- saveotufillphylum[saveotufillphylum$otu%in%saveotu8$otu,]

# Create side-by-side box plots
show_col(viridis_pal(option = "magma")(20))
```

```
#dark_purple1 => "#400F73FF"
#dark_pruple2 => "#681D81FF"      => RL
#organge1 => "#FD9A6AFF"
#orange2 => "#FEB37BFF"           => SL
#Red => "D6456CFF"                => Soylec
#red_purple => "AB337CFF"         => Control


#magma colors
# EM_Colors <- c("#837239B3", "#AB337CFF","#cecb99","#681D81FF", "#FEB37BFF", "#D6456CFF")
# #                CMC           Control    tween80    RL          SL      SoyLec

X.order.sign <- c("Control - 0","P80 - 0.05","P80 - 0.5","RL - 0.05","RL - 0.5")
# EM_Colors.sign <- c("#cecb99","#5d6295B3", "#c9e5ff", "#953d6aB3")

#original colors
EM_Colors <- c("#999999","#A660CC", "#4EAC57", "#E1A13F", "#A660CC", "#21AADE")
#                CMC           Control    tween80    RL          SL      SoyLec

#CD5C5C = red
#4EAC57 = green
#21AADE = blue
#A660CC = purple
```

```
#E1A13F = orange
#999999 = grey

plot_G1 <- ggplot(savegenus1,aes(x=Treatment,y=count,fill=Emulsifier))+
  geom_boxplot()+
  #geom_point(data=saveotu1[log(saveotu1$count)>saveotu1$upper.limit|
  # log(saveotu1$count)<saveotu1$lower.limit,],aes(x=Treatment,y=count,color=Treatment))+
  scale_x_discrete(limits=X.order.sign)

plot_G1 <- plot_G1 +
  scale_y_continuous(labels=function(x) format(x,scientific=TRUE,digits=1),trans='log10',
                     breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free', labeller = label_wrap_gen(20))

plot_G1 <- papertheme(plot_G1)+
  #theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=NC_PD_colors,name=NULL)+
  #theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"),
        axis.title.x = element_blank())+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),size=3, color = "black")
plot_G1
```

```
mylegenddeseq <- get_legend(plot_G1)

# deseqsaveotu1 <-plot_G1+ theme(legend.position = "none",
# axis.line = element_line(color="black"))
# deseqsaveotu1strips <- ggplot_gtable(ggplot_build(deseqsaveotu1))
# stripr <- which(grepl('strip-t',deseqsaveotu1strips$layout$name))
#checks if "strip-t" can be found in any of the names in the layout in deseqsaveotu1strips
# fills <- as.character(saveotufillphylum1$fill)
# k <- 1
# for (i in stripr) {
#   j <- which(grepl('rect', deseqsaveotu1strips$grobs[[i]]$grobs[[1]]$childrenOrder))
#   deseqsaveotu1strips$grobs[[i]]$grobs[[1]]$children[[j]]$gp$fill <- fills[k]
#   k <- k+1
# }
# grid.draw(deseqsaveotu1strips)


plot_G2 <- ggplot(savegenus2,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G2 <- plot_G2 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free', labeller = label_wrap_gen(20))
```

```
plot_G2 <- papertheme(plot_G2)+theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10), color = "black")
plot_G2
```

```
savegenus3$genus <- mapvalues(savegenus3$genus, from = "Candidatus_Soleaferrea",
                              to = "Candidatus Soleaferrea")
savegenus3$otu <- mapvalues(savegenus3$otu, from = "Candidatus_Soleaferrea",
                            to = "Candidatus Soleaferrea")

savegenus3$genus <- mapvalues(savegenus3$genus, from = "Clostridia_unclassified",
                              to = "Clostridia unclassified")
savegenus3$otu <- mapvalues(savegenus3$otu, from = "Clostridia_unclassified",
                            to = "Clostridia unclassified")

plot_G3 <- ggplot(savegenus3,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G3 <- plot_G3 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                           labeller = labeller(otu = label_wrap_gen(20)))

plot_G3 <- papertheme(plot_G3)+theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none")+
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y =  element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10), color = "black")
plot_G3
```

```
savegenus4$genus <- mapvalues(savegenus4$genus, from = "Desulfovibrionaceae_unclassified",
                              to = "Desulfovibrionaceae unclassified")
savegenus4$otu <- mapvalues(savegenus4$otu, from = "Desulfovibrionaceae_unclassified",
                            to = "Desulfovibrionaceae unclassified")

plot_G4 <- ggplot(savegenus4,aes(x=Treatment,y=count,fill=Emulsifier)) + geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G4 <- plot_G4 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
```

```
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G4 <- papertheme(plot_G4) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL)+
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
  theme(legend.position = "none") +
  theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
        axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10), color = "black")
plot_G4
```

```
savegenus5$genus <- mapvalues(savegenus5$genus, from = "Enterobacterales_unclassified",
                              to = "Enterobacterales unclassified")
savegenus5$otu <- mapvalues(savegenus5$otu, from = "Enterobacterales_unclassified",
                            to = "Enterobacterales unclassified")

savegenus5$genus <- mapvalues(savegenus5$genus, from = "Enterobacteriaceae_unclassified",
                              to = "Enterobacteriaceae unclassified")
savegenus5$otu <- mapvalues(savegenus5$otu, from = "Enterobacteriaceae_unclassified",
                            to = "Enterobacteriaceae unclassified")

savegenus5$genus <- mapvalues(savegenus5$genus, from = "Erysipelotrichaceae_ge",
                              to = "Erysipelotrichaceae ge")
savegenus5$otu <- mapvalues(savegenus5$otu, from = "Erysipelotrichaceae_ge",
                            to = "Erysipelotrichaceae ge")


plot_G5 <- ggplot(savegenus5,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G5 <- plot_G5 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G5 <- papertheme(plot_G5) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),
      axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G5
```

```
savegenus6$genus <- mapvalues(savegenus6$genus, from = "Gastranaerophilales_ge",
                              to = "Gastranaerophilales ge")
```

```
savegenus6$otu <- mapvalues(savegenus6$otu, from = "Gastranaerophilales_ge",
                            to = "Gastranaerophilales ge")

plot_G6 <- ggplot(savegenus6,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G6 <- plot_G6 +
  scale_y_continuous(labels=function(x) format(x,scientific=TRUE,digits=1),
                     trans='log10',breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") + facet_grid(.~otu,drop=TRUE,scales='free',
                                                         labeller = label_wrap_gen(20))

plot_G6 <- papertheme(plot_G6) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white"))+
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G6
```

```
savegenus7$genus <- mapvalues(savegenus7$genus, from = "Lachnospiraceae_ge",
                              to = "Lachnospiraceae ge")
savegenus7$otu <- mapvalues(savegenus7$otu, from = "Lachnospiraceae_ge",
                            to = "Lachnospiraceae ge")

plot_G7 <- ggplot(savegenus7,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G7 <- plot_G7 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G7 <- papertheme(plot_G7) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),
      axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G7
```

```
savegenus8$genus <- mapvalues(savegenus8$genus, from = "Lachnospiraceae_ND3007_group",
                              to = "Lachnospiraceae ND3007 group")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Lachnospiraceae_ND3007_group",
                            to = "Lachnospiraceae ND3007 group")
```

```
savegenus8$genus <- mapvalues(savegenus8$genus, from = "Lachnospiraceae_unclassified",
                              to = "Lachnospiraceae unclassified")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Lachnospiraceae_unclassified",
                            to = "Lachnospiraceae unclassified")

savegenus8$genus <- mapvalues(savegenus8$genus, from = "Lactobacillales_unclassified",
                              to = "Lactobacillales unclassified")
savegenus8$otu <- mapvalues(savegenus8$otu, from = "Lactobacillales_unclassified",
                            to = "Lactobacillales unclassified")

plot_G8 <- ggplot(savegenus8,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G8 <- plot_G8 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G8 <- papertheme(plot_G8) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G8
```

```
savegenus9$genus <- mapvalues(savegenus9$genus, from = "Micrococcaceae_unclassified",
                              to = "Micrococcaceae unclassified")
savegenus9$otu <- mapvalues(savegenus9$otu, from = "Micrococcaceae_unclassified",
                            to = "Micrococcaceae unclassified")

plot_G9 <- ggplot(savegenus9,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G9 <- plot_G9 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                  digits=1),
                                        trans='log10',
                                        breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G9 <- papertheme(plot_G9) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
```

```
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G9
```

```
savegenus10$genus <- mapvalues(savegenus10$genus, from = "Oscillospiraceae_unclassified",
                               to = "Oscillospiraceae unclassified")
savegenus10$otu <- mapvalues(savegenus10$otu, from = "Oscillospiraceae_unclassified",
                             to = "Oscillospiraceae unclassified")

savegenus10$genus <- mapvalues(savegenus10$genus, from = "Oscillospirales_unclassified",
                               to = "Oscillospirales unclassified")
savegenus10$otu <- mapvalues(savegenus10$otu, from = "Oscillospirales_unclassified",
                             to = "Oscillospirales unclassified")

plot_G10 <- ggplot(savegenus10,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G10 <- plot_G10 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                    digits=1),
                                          trans='log10',
                                          breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G10 <- papertheme(plot_G10) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G10
```

```
savegenus11$genus <- mapvalues(savegenus11$genus, from = "Prevotellaceae_unclassified",
                               to = "Prevotellaceae unclassified")
savegenus11$otu <- mapvalues(savegenus11$otu, from = "Prevotellaceae_unclassified",
                             to = "Prevotellaceae unclassified")

savegenus11$genus <- mapvalues(savegenus11$genus, from = "Ruminococcaceae_unclassified",
                               to = "Ruminococcaceae unclassified")
savegenus11$otu <- mapvalues(savegenus11$otu, from = "Ruminococcaceae_unclassified",
                             to = "Ruminococcaceae unclassified")

plot_G11 <- ggplot(savegenus11,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G11 <- plot_G11 +
  scale_y_continuous(labels=function(x) format(x,scientific=TRUE,digits=1),
                     trans='log10',
                     breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
```

```
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G11 <- papertheme(plot_G11) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G11
```

```
savegenus12$genus <- mapvalues(savegenus12$genus, from = "Selenomonadaceae_unclassified",
                               to = "Selenomonadaceae unclassified")
savegenus12$otu <- mapvalues(savegenus12$otu, from = "Selenomonadaceae_unclassified",
                             to = "Selenomonadaceae unclassified")

savegenus12a <- dplyr::filter(savegenus12,genus %in% c("Selenomonadaceae unclassified",
                                                       "Subdoligranulum"))


plot_G12a <- ggplot(savegenus12a,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G12a <- plot_G12a + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                      digits=1),
                                            trans='log10',
                                          breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G12a <- papertheme(plot_G12a) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.text.x = element_blank(),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G12a
```

```
savegenus12b <- dplyr::filter(savegenus12,genus %in% c("Tyzzerella","UBA1819","UCG-003"))

plot_G12b <- ggplot(savegenus12b,aes(x=Treatment,y=count,fill=Emulsifier)) + geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G12b <- plot_G12b + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                      digits=1),
                                            trans='log10',
                                          breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))
```

```
plot_G12b <- papertheme(plot_G12b) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),axis.line.y = element_blank(),
      axis.ticks.y = element_blank(), axis.text.x = element_text(angle = 90, vjust = 0.5,
                                                                 hjust = 1),
      axis.text.y= element_blank(),axis.title.y = element_blank()) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G12b
```

```
savegenus13$genus <- mapvalues(savegenus13$genus, from = "uncultured_ge",
                               to = "uncultured ge")
savegenus13$otu <- mapvalues(savegenus13$otu, from = "uncultured_ge",
                             to = "uncultured ge")

plot_G13 <- ggplot(savegenus13,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot()+
   scale_x_discrete(limits=X.order.sign)
  #geom_point(data=saveotu2[log(saveotu2$count)>saveotu2$upper.limit|
#log(saveotu2$count)<saveotu2$lower.limit,],aes(x=Treatment,y=count,color=Treatment))
plot_G13 <- plot_G13 + scale_y_continuous(labels=function(x) format(x,scientific=TRUE,
                                                                    digits=1),
                                          trans='log10',
                                          breaks = trans_breaks('log10', function(x) 10^x))+
  ylab("Normalised log transformed counts") +
  facet_grid(.~otu,drop=TRUE,scales='free',labeller = label_wrap_gen(20))

plot_G13 <- papertheme(plot_G13) + theme(axis.title.x = element_blank()) +
  scale_fill_manual(values=EM_Colors,name=NULL) +
  #scale_color_manual(values=EM_Colors,name=NULL)+
# theme(legend.justification = "center") +
theme(legend.position = "none")+
theme(strip.text.x=element_text(size=14),
      axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1),
      axis.text.y= element_text(size = 14),axis.title.y = element_text(color="white")) +
  geom_text(aes(label=label,x=Treatment,y=max(count)+10),color = "black")
plot_G13
```

```
plot_X <- ggplot(savegenus13,aes(x=Treatment,y=count,fill=Emulsifier))+ geom_boxplot() +
  scale_x_discrete(limits=X.order.sign, drop=TRUE)+
  scale_fill_manual(breaks=c("Control","P80","RL"),values=c("#999999","#E1A13F","#4EAC57",
                                                            "#A660CC","#21AADE"),
                    name="Emulsifier")+
  guides(fill = guide_legend(reverse = TRUE))+
  theme(legend.text = element_text(size=12,family='sans',angle=0),
        legend.title = element_text(size=14,family='sans',angle=0))
plot_X
```

```
mylegenddeseq <- get_legend(plot_X)

#original colors
EM_Colors <- c("#CD5C5C", "#999999", "#4EAC57", "#E1A13F", "#A660CC", "#21AADE")
#                CMC           Control    tween80    RL          SL      SoyLec
```

```
#EM_Colors <- c("#837239B3", "#557d7aB3","#cecb99","#5d6295B3", "#c9e5ff", "#953d6aB3")
                  #brown     #green    #yellow     #purple    #light blue   #red
                  #CMC        #Control    #P80        #RL         #SL       SoyLec

# Multi plot of bargraphs
titledeseq1 <- ggdraw() + draw_label("Normalised log transformed counts",angle=90,size=20)

deseqgenusotuselect1 <- ggdraw() + draw_plot(plot_G1,0,0.8666666,1,0.13) +
  draw_plot(plot_G2,0,0.73333333,1,0.13) + draw_plot(plot_G3,0,0.60000,1,0.13) +
  draw_plot(plot_G4,0,0.466666,1,0.13) + draw_plot(plot_G5,0,0.3333333,1,0.13) +
  draw_plot(plot_G6,0,0.2,1,0.13) + draw_plot(plot_G7,0,0.02,1,0.18) +
  draw_plot(mylegenddeseq,0,0,1,0.02)

tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_DESeq_Genera_Barplot_
     width=12*dpi,height=15*dpi,res=dpi,compression="lzw")
plot_grid(titledeseq1,deseqgenusotuselect1,ncol=2,rel_widths=c(0.02,0.98))
dev.off()


titledeseq2 <- ggdraw() + draw_label("Normalised log transformed counts",angle=90,size=20)

deseqgenusotuselect2 <- ggdraw() + draw_plot(plot_G8,0,0.84,1,0.16) +
  draw_plot(plot_G8,0,0.68,1,0.16) + draw_plot(plot_G10,0,0.52,1,0.16) +
  draw_plot(plot_G11,0,0.36,1,0.16) + draw_plot(plot_G12a,0,0.2,0.45,0.16) +
  draw_plot(plot_G12b,0.45,0.14,0.55,0.22) + draw_plot(plot_G13,0,0.02,0.45,0.18) +
  draw_plot(mylegenddeseq,0,0,1,0.02)

tiff("/media/projects2/LisaM/SHIME_EM/Illumina/SHIME_16S_Processing/Figures/SHIME_DESeq_Genera_Barplot_
     width=12*dpi,height=15*dpi,res=dpi,compression="lzw")
plot_grid(titledeseq2,deseqgenusotuselect2,ncol=2,rel_widths=c(0.02,0.98))
dev.off()

# Get Log2FoldChange
Log2FoldChange <- data.frame()

for (i in genusnames.P80.RL.mucus){


LFCP80_0.05_contr_mucus_G <- resP80_0.05_contr_mucus_G[i,]$log2FoldChange
names(LFCP80_0.05_contr_mucus_G) <- c("P80_0.05-contr")
LFCP80_0.5_contr_mucus_G <- resP80_0.5_contr_mucus_G[i,]$log2FoldChange
names(LFCP80_0.5_contr_mucus_G) <- c("P80_0.5-contr")


LFCRL_0.05_contr_mucus_G <- resRL_0.05_contr_mucus_G[i,]$log2FoldChange
names(LFCRL_0.05_contr_mucus_G) <- c("RL_0.05-contr")
LFCRL_0.5_contr_mucus_G <- resRL_0.5_contr_mucus_G[i,]$log2FoldChange
names(LFCRL_0.5_contr_mucus_G) <- c("RL_0.5-contr")

L2FCtotal <- c(LFCP80_0.05_contr_mucus_G,LFCP80_0.5_contr_mucus_G,LFCRL_0.05_contr_mucus_G,LFCRL_0.5_co

Log2FoldChange <- rbind(Log2FoldChange,L2FCtotal)
}
```

```
colnames(Log2FoldChange)[1:4] <- X.order[1:4]
Log2FoldChange$Genus <- genusnames.P80.RL.mucus
write.xlsx(Log2FoldChange,"Deseq_Log2FoldChange_P80_RL_Mucus.xlsx")
```