BetaDiversity_Analysis

Before you begin:

These scripts were tailored for the analyses performed in:

Seibert et al, 2021, *Mild and severe SARS-CoV-2 infection induces respiratory and intestinal microbiome changes in the K18-hACE2 transgenic mouse model*

Purpose:

The purpose of this script is to analyze beta diversity of the different groups within the ceca and the lungs. This will correspond to figures 2D-G, 4D and 4H, and 5D-G

Load the needed packages

```
library(eulerr)
library(ggplot2)
library(ggpubr)
library(phyloseq)
library(vegan)
library(dplyr)
library(microbiome)
```

Import the metadata needed for this analysis from the alpha diversity analysis

```
sampledataveg <- readRDS("/Users/sampledataveg.rds")
```

Figure 2D: Venn Diagram of ceca samples

To examine the similarities between the groups lets look at how many ASVs the groups have in common with one another

Load the R file from previous data processing

```
# Import PhyloSeq object with only the rarified cecum data from the PBS/Vehicle, Low/Vehicle,
High/Vehicle dose group from alpha diversity analysis
Cecum.group1 <- readRDS("/Users/Cecum.group1.rds")

# Create a table with rarified count data
table(meta(Cecum.group1)$Group, useNA = "always")

# Convert to relative abundances
cecum.rel <- microbiome::transform(Cecum.group1, "compositional")

# Make a list of the Groups
group_states <- unique(as.character(meta(Cecum.group1)$Group))
print(group_states)

# Write a for loop to go through each of the disease_states one by one and combine identified core taxa
into a list.
list_core <- c() # an empty object to store information

for (n in group_states){ # for each variable n in group
    #print(paste0("Identifying Core Taxa for ", n))

    ps.sub <- subset_samples(Cecum.group1, Group == n) # Choose sample from group by n

    core_m <- core_members(ps.sub, # ps.sub is phyloseq selected with only samples from g
                           detection = 0.001, # 0.001 in atleast 90% samples
                           include.lowest = TRUE)
                           #prevalence = 0.75)
    print(paste0("No. of core taxa in ", n, " : ", length(core_m))) # print core taxa identified in
each DiseaseState.
    list_core[[n]] <- core_m # add to a list core taxa for each group.
    #print(list_core)
}

# Specify colors and plot venn
# Supplying colors in the order they appear in list_core
```

```
mycols <- c("grey","orange","steelblue2")
plot(venn(list_core),
     fills = mycols)
```

Figure 2E: Calculate Bray-Curtis distances and compare distances using boxplots in the ceca

First, make a matrix of the cecum counts

```
# Import the file of the rarified ASVs that is from the alpha diversity analysis
sample.rare <- readRDS(file = "/Users/RarifiedASVs.rds")

# List of the row names that correspond to lungs or other groups that we are not analyzing in the cecum
remove= c("BS.204", "BS.205", "BS.206", "BS.207", "BS.208", "BS.209", "BS.210", "BS.211", "BS.212",
"BS.213", "BS.214", "BS.215", "BS.216", "BS.217", "BS.218", "BS.219", "BS.220", "BS.221", "BS.222",
"BS.223", "BS.224", "BS.225", "BS.226", "BS.227", "BS.228", "BS.229", "BS.230", "BS.231", "BS.232",
"BS.233", "BS.234", "BS.235", "BS.236", "BS.237", "BS.238","BS.239", "BS.240", "BS.241", "BS.242",
"BS.243", "BS.250", "BS.251", "BS.252", "BS.253", "BS.269", "BS.270", "BS.271", "BS.272")

# Remove the row names listed in the rarified ASV file
sample.rare.cecum = sample.rare[!row.names(sample.rare)%in%remove,]

# Specify the number of rows aka samples left to check that it worked
nrow(sample.rare.cecum)
```

Next, calculate the bray-curtis distances and graph in boxplots

```
# I will calculate the bray-curtis distances
bray_cecum_vehicle <- vegdist(sample.rare.cecum, method = "bray", binary = FALSE)

# Turn the distance class into a matrix
bray_cecum_vehicle_matrix <- as.matrix(bray_cecum_vehicle)

# Export that matrix into a csv file
write.csv(bray_cecum_vehicle_matrix, "bray_cecum_vehicle_matrix.csv")

# Convert the matrix into a long format by hand
# I tried using a program but it omitted samples so I decided to do it by hand
bray_cecum.melt2.sd <- read.csv("/Users/bray_cecum_distance.csv")

# Reorder the groups so that they are in the same order
bray_cecum.melt2.sd$Group1<- factor(bray_cecum.melt2.sd$Group1, levels = c('PBS', 'Low', 'High'))
bray_cecum.melt2.sd$Group2<- factor(bray_cecum.melt2.sd$Group2, levels = c('PBS', 'Low', 'High'))

# Plot the groups
ggplot(bray_cecum.melt2.sd, aes(x = Group2, y = Distance)) +
  theme_bw() +
  theme(panel.grid = element_blank())+
  #geom_point() +
  geom_boxplot(aes(color = ifelse(Group1 == Group2,"goldenrod3", "black"))) +
  scale_color_identity() +
  facet_wrap(~ Group1, scales = "free_x") +
  theme(axis.text.x=element_text(hjust = 1, vjust = 0.5)) +
  scale_y_continuous(breaks = seq(0, 1, by=0.2), limits=c(0, 1))+
  ggtitle(paste0("Distance Metric = ", "Bray-Curtis")) +
  ylab("Bray-Curtis Distance") +
  xlab("Group")

# Statistics testing for comparison among means of the bray curtis for different groups (this uses the
Wilcoxon Sign test (non-parametric))

# Look at PBS group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "PBS")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)

# Look at Low group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "Low")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)

# Look at High group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "High")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)
```

Figure 2F: Visualize Bray-Curtis distances using NMDS plot and calculate the PERMANOVA for difference of groups in the cecum

```
# I will calculate the bray-curtis distances
# The Wisconsin transformation normalizes 0-1 so it will be very similar to the unweighted so i will
not use the transformation
sample.nmds <- metaMDS(sample.rare.cecum, distance="bray", try=20, trymax=100, autotransform=FALSE)

# Plot the bray-curtis distances in simple plot
ordiplot(sample.nmds, disp="sites")

# For more control of what your plot looks like in ggplot
sample.bray.points <- scores(sample.nmds, choices=c(1,2), disp="sites") #extract site scores for all
sites

# Duplicate the SampleID column
sample.bray.points <- cbind(rownames(sample.bray.points), data.frame(sample.bray.points,
row.names=NULL))
colnames(sample.bray.points)[1] <- "SampleID"

# Merge the bray-curtis data with the sample metadata by SampleID
sample.bray.pointsMerged <- merge(sample.bray.points, sampledataveg, by = "SampleID")

# Reorder the groups
sample.bray.pointsMerged$Group<- factor(sample.bray.pointsMerged$Group, levels = c('PBS', 'Infected-3-
Vehicle', 'Infected-5-Vehicle'))

# Set the colors for the graph
colorgroups = c("black","darkorange2", "blue")

# Plot the graph using ggplot
p1 <-ggplot(sample.bray.pointsMerged, aes(x = NMDS1, y = NMDS2, fill = Group, shape = as.factor(dpi)))
+
geom_point(aes(size = 1.5)) +
stat_ellipse(aes(color = Group,group = Group), alpha = 0.1)+
scale_fill_manual(values = colorgroups)+
scale_color_manual(values = colorgroups)+
scale_shape_manual(values = c(21:26)) +
theme_bw() +
theme(panel.grid = element_blank())+
theme(strip.text.y = element_text(angle = 0))+
guides(fill=guide_legend(override.aes=list(shape=21)))
p1

# PERMANOVA ANALYSIS
pseq.rel <- microbiome::transform(Cecum.group1, "compositional")
otu <- abundances(pseq.rel)
meta <- meta(pseq.rel)

permanova <- adonis(t(otu) ~ Group,
                    data = meta, permutations=1000, method = "bray")
permanova
```

Figure 2G: Visualize Bray-Curtis distances using a dendrogram in the ceca

```
# I will calculate the bray-curtis distances
bray_cecum_vehicle <- vegdist(sample.rare.cecum, method = "bray", binary = FALSE)

clust.res <- hclust(bray_cecum_vehicle, method = "average")

# Calculate cophenetic correalation coefficient
d2 <- cophenetic(clust.res)

# Compare original distance matrix with the cophenetic correalation coefficient
cor(bray_cecum_vehicle, d2)

# Plot the dendogram
plot(clust.res, hang = -1)
```

There are also methods for evaluating each linkage method that i will use for hclust. One method is called cophenetic correalation coefficient (CCC). It is a Pearson correlation between original distance matrix and cophenetic distances matrix of dendrogram (cluster configuration). Higher

values of CCC (nearer to 1) = better clusterization (usually values above 0.75 are considered good).

Results
- ward.D has a correlation of 0.78
- D2 has a correlation of 0.78
- single has a correlation of 0.75
- complete has a correlation of 0.78
- average (UPGMA) has a correlation of 0.79
- mcquitty (WPGMA) has a correlation of 0.73
- median (WPGMC) has a correlation of 0.69
- centroid (UPGMC) has a correlation of 0.65

I will use UPGMA or average since it has the highest correlation

In the dendrogram, the y-axis is simply the value of this distance metric between clusters. For example, if you see two clusters merged at a height x, it means that the distance between those clusters was x

The colors of groups and dpc were added in illustrator.

Figure 4D: Bray-Curtis NMDS of low doses treated and not treated with GC-376

Lets look at the alpha diversity graphs for Vehicle versus GC-376 in lungs from groups that were infected with a low virus dose to investigate if there is an antiviral effect on the lung microbiome

First, make a matrix of the lung counts

```
# List of the row names that correspond to the cecum that we are not analyzing in the lungs
remove= c("BS.244", "BS.245", "BS.246", "BS.247", "BS.248", "BS.249", "BS.250", "BS.251", "BS.252",
"BS.253", "BS.254", "BS.255", "BS.256", "BS.257", "BS.258", "BS.259", "BS.260", "BS.261", "BS.262",
"BS.263", "BS.264", "BS.265", "BS.266", "BS.267", "BS.268", "BS.269", "BS.270", "BS.271", "BS.272") #
list of rownames I would like to remove from file "data"

sample.rare.lung = sample.rare[!row.names(sample.rare)%in%remove,]
nrow(sample.rare.lung)

# List of the row names that correspond to the lungs collected from mice challenged with a high dose
since we want to first look at the low dose
remove= c("BS.204", "BS.205", "BS.206", "BS.207", "BS.208", "BS.209", "BS.210", "BS.211", "BS.212",
"BS.222", "BS.223", "BS.224", "BS.225", "BS.226", "BS.227", "BS.237", "BS.238", "BS.239", "BS.240",
"BS.241", "BS.242", "BS.243") # list of rownames I would like to remove from file "data"

# Remove the row names listed in the rarified ASV file
sample.rare.lung.inf3 = sample.rare.lung[!row.names(sample.rare.lung)%in%remove,]

# Specify the number of rows aka samples left to check that it worked
nrow(sample.rare.lung.inf3)
```

Visualize Bray-Curtis distances using NMDS plot and calculate the PERMANOVA for difference of groups in the cecum

```
# I will calculate the bray-curtis distances
# The Wisconsin transformation normalizes 0-1 so it will be very similar to the unweighted so i will
not use the transformation
sample.nmds <- metaMDS(sample.rare.lung.inf3, distance="bray", try=20, trymax=100, autotransform=FALSE)

# Plot the bray-curtis distances in simple plot
ordiplot(sample.nmds, disp="sites")

# For more control of what your plot looks like in ggplot
sample.bray.points <- scores(sample.nmds, choices=c(1,2), disp="sites") #extract site scores for all
sites

# Duplicate the SampleID column
sample.bray.points <- cbind(rownames(sample.bray.points), data.frame(sample.bray.points,
row.names=NULL))
colnames(sample.bray.points)[1] <- "SampleID"

# Merge the bray-curtis data with the sample metadata by SampleID
sample.bray.pointsMerged <- merge(sample.bray.points, sampledataveg, by = "SampleID")
```

```
# Reorder the groups
sample.bray.pointsMerged$Group<- factor(sample.bray.pointsMerged$Group, levels = c('Infected-3-
Vehicle', 'Infected-3-GC376'))

# Set the colors for the graph
colorgroups = c("darkorange2","forestgreen")

# Plot the graph using ggplot
ggplot(sample.bray.pointsMerged, aes(x = NMDS1, y = NMDS2, fill = Group, shape = as.factor(dpi))) +
geom_point(aes(size = 1.5))+
scale_shape_manual(values = c(21,21,22,22,23,23))+
scale_fill_manual(values = colorgroups)+
stat_ellipse(aes(color = Group, group = Group), alpha = 0.3)+
scale_fill_manual(values = colorgroups)+
scale_color_manual(values = colorgroups)+
theme_bw() +
theme(panel.grid = element_blank())+
theme(strip.text.y = element_text(angle = 0))+
guides(fill=guide_legend(override.aes=list(shape=21)))

# PERMANOVA ANALYSIS

# Import PhyloSeq object with only the rarified lung data from the low dose group from alpha diversity
analysis
Lung.group.3 <- readRDS("/Users/Lung.group.3.rds")

# Calculate the PERMANOVA
pseq.rel <- microbiome::transform(Lung.group.3, "compositional")
otu <- abundances(pseq.rel)
meta <- meta(pseq.rel)

permanova <- adonis(t(otu) ~ Group,
              data = meta, permutations=1000, method = "bray")
permanova
```

Figure 5S-A: Bray-Curtis boxplots of low doses treated and not treated with GC-376

```
# I will calculate the bray-curtis distances
bray_lung_vehicle_3 <- vegdist(sample.rare.lung.inf3, method = "bray", binary = FALSE)

# Turn the distance class into a matrix
bray_lung_vehicle_3_matrix <- as.matrix(bray_lung_vehicle_3)

# Export that matrix into a csv file
write.csv(bray_lung_vehicle_3_matrix, "bray_lung_vehicle_3_matrix.csv")

# Convert the matrix into a long format by hand
# I tried using a program but it omitted samples so I decided to do it by hand
bray_lung_3.melt2.sd <- read.csv("/Users/bray_lung_vehicle_3_distance.csv")

# Reorder the groups so that they are in the same order
bray_lung_3.melt2.sd$Group1<- factor(bray_lung_3.melt2.sd$Group1, levels = c('Vehicle', 'GC376'))
bray_lung_3.melt2.sd$Group2<- factor(bray_lung_3.melt2.sd$Group2, levels = c('Vehicle', 'GC376'))

# Plot the graph using ggplot
ggplot(bray_lung_3.melt2.sd, aes(x = Group2, y = Distance)) +
  theme_bw() +
  theme(panel.grid = element_blank())+
  #geom_point() +
  geom_boxplot(aes(color = ifelse(Group1 == Group2,"goldenrod3", "black"))) +
  scale_color_identity() +
  facet_wrap(~ Group1, scales = "free_x") +
  theme(axis.text.x=element_text(hjust = 1, vjust = 0.5)) +
  scale_y_continuous(breaks = seq(0, 1, by=0.2), limits=c(0, 1))+
  ggtitle(paste0("Distance Metric = ", "Bray-Curtis")) +
  ylab("Bray-Curtis Distance") +
  xlab("Group")

# Statistics testing for comparison among means of the bray curtis for different groups (this uses the
Wilcoxon Sign test (non-parametric))
# Look at Vehicle group only
bray_lung.melt2.filter <- bray_lung_3.melt2.sd %>%
  filter(Group1 == "Vehicle")
compare_means(Distance ~ Group2,  data = bray_lung.melt2.filter)
```

```
# Look at GC376 group only
bray_lung.melt2.filter <- bray_lung_3.melt2.sd %>%
  filter(Group1 == "GC376")
compare_means(Distance ~ Group2,  data = bray_lung.melt2.filter)
```

Figure 4H: Bray-Curtis NMDS of high doses treated and not treated with GC-376

Lets look at the alpha diversity graphs for Vehicle versus GC-376 in lungs from groups that were infected with a high virus dose to investigate if there is an antiviral effect on the lung microbiome

First, make a matrix of the lung counts

```
# List of the row names that correspond to the lungs collected from mice challenged with a low dose since we want to first look at the low dose
remove= c("BS.204", "BS.205", "BS.206", "BS.207", "BS.208", "BS.209", "BS.210", "BS.211", "BS.212", "BS.213", "BS.214", "BS.215", "BS.216", "BS.217", "BS.218", "BS.219", "BS.220", "BS.221", "BS.228", "BS.229", "BS.230", "BS.231", "BS.232", "BS.233", "BS.234", "BS.235", "BS.236") # list of rownames I would like to remove from file "data"

# Remove the row names listed in the rarified ASV file
sample.rare.lung.inf5 = sample.rare.lung[!row.names(sample.rare.lung)%in%remove,]

# Specify the number of rows aka samples left to check that it worked
nrow(sample.rare.lung.inf5)
```

Visualize Bray-Curtis distances using NMDS plot and calculate the PERMANOVA for difference of groups in the cecum

```
# I will calculate the bray-curtis distances
# The Wisconsin transformation normalizes 0-1 so it will be very similar to the unweighted so i will not use the transformation
sample.nmds <- metaMDS(sample.rare.lung.inf5, distance="bray", try=20, trymax=100, autotransform=FALSE)

# Plot the bray-curtis distances in simple plot
ordiplot(sample.nmds, disp="sites")

# For more control of what your plot looks like in ggplot
sample.bray.points <- scores(sample.nmds, choices=c(1,2), disp="sites") #extract site scores for all sites

# Duplicate the SampleID column
sample.bray.points <- cbind(rownames(sample.bray.points), data.frame(sample.bray.points, row.names=NULL))
colnames(sample.bray.points)[1] <- "SampleID"

# Merge the bray-curtis data with the sample metadata by SampleID
sample.bray.pointsMerged <- merge(sample.bray.points, sampledataveg, by = "SampleID")

# Reorder the groups
sample.bray.pointsMerged$Group<- factor(sample.bray.pointsMerged$Group, levels = c('Infected-5-Vehicle', 'Infected-5-GC376'))

# Set the colors for the graph
colorgroups = c("blue", "maroon1")

# Plot the graph using ggplot
ggplot(sample.bray.pointsMerged, aes(x = NMDS1, y = NMDS2, fill = Group, shape = as.factor(dpi))) +
geom_point(aes(size = 1.5))+
scale_shape_manual(values = c(21,21,22,22,23,23))+
scale_fill_manual(values = colorgroups)+
stat_ellipse(aes(color = Group, group = Group), alpha = 0.3)+
scale_fill_manual(values = colorgroups)+
scale_color_manual(values = colorgroups)+
theme_bw() +
theme(panel.grid = element_blank())+
theme(strip.text.y = element_text(angle = 0))+
guides(fill=guide_legend(override.aes=list(shape=21)))

# PERMANOVA Analysis

# Import PhyloSeq object with only the rarified lung data from the high dose group from alpha diversity analysis
Lung.group.5 <- readRDS("/Users/Lung.group.5.rds")
```

```
# Calculate the PERMANOVA
pseq.rel <- microbiome::transform(Lung.group.5, "compositional")
otu <- abundances(pseq.rel)
meta <- meta(pseq.rel)

permanova <- adonis(t(otu) ~ Group,
                data = meta, permutations=1000, method = "bray")
permanova
```

Figure 5S: Bray-Curtis boxplots of High doses treated and not treated with GC-376

```
# I will calculate the bray-curtis distances
bray_lung_vehicle_5 <- vegdist(sample.rare.lung.inf5, method = "bray", binary = FALSE)

# Turn the distance class into a matrix
bray_lung_vehicle_5_matrix <- as.matrix(bray_lung_vehicle_5)

# Export that matrix into a csv file
write.csv(bray_lung_vehicle_5_matrix, "bray_lung_vehicle_5_matrix.csv")

# Convert the matrix into a long format by hand
# I tried using a program but it omitted samples so I decided to do it by hand
bray_lung_5.melt2.sd <- read.csv("/Users/bray_lung_vehicle_5_distance.csv")

# Reorder the groups so that they are in the same order
bray_lung_5.melt2.sd$Group1<- factor(bray_lung_3.melt2.sd$Group1, levels = c('Vehicle', 'GC376'))
bray_lung_5.melt2.sd$Group2<- factor(bray_lung_3.melt2.sd$Group2, levels = c('Vehicle', 'GC376'))

# Plot the graph using ggplot
ggplot(bray_lung_5.melt2.sd, aes(x = Group2, y = Distance)) +
  theme_bw() +
  theme(panel.grid = element_blank())+
  #geom_point() +
  geom_boxplot(aes(color = ifelse(Group1 == Group2,"goldenrod3", "black"))) +
  scale_color_identity() +
  facet_wrap(~ Group1, scales = "free_x") +
  theme(axis.text.x=element_text(hjust = 1, vjust = 0.5)) +
  scale_y_continuous(breaks = seq(0, 1, by=0.2), limits=c(0, 1))+
  ggtitle(paste0("Distance Metric = ", "Bray-Curtis")) +
  ylab("Bray-Curtis Distance") +
  xlab("Group")

# Statistics testing for comparison among means of the bray curtis for different groups (this uses the
Wilcoxon Sign test (non-parametric))
# Look at Vehicle group only
bray_lung.melt2.filter <- bray_lung_5.melt2.sd %>%
  filter(Group1 == "Vehicle")
compare_means(Distance ~ Group2,  data = bray_lung.melt2.filter)

# Look at GC376 group only
bray_lung.melt2.filter <- bray_lung_5.melt2.sd %>%
  filter(Group1 == "GC376")
compare_means(Distance ~ Group2,  data = bray_lung.melt2.filter)
```

Figure 5D: Venn Diagram of Lungs treated with GC-376

To examine the similarities between the groups lets look at how many ASVs the groups have in common with one another

Load the R file from previous data processing

```
# Import PhyloSeq object with only the rarified lung data from the Mock/GC-376, Low/GC-376, High/GC-376
dose group from alpha diversity analysis
Lung.group.GC376 <- readRDS("/Users/baseibert/Lung.group.GC376.rds")
# Use phyloseq object with rarified count data
table(meta(Lung.group.GC376)$Group, useNA = "always")

# Convert to relative abundances
lung.rel <- microbiome::transform(Lung.group.GC376, "compositional")

# Make a list of the Groups
group_states <- unique(as.character(meta(Lung.group.GC376)$Group))
print(group_states)
```

```
# Write a for loop to go through each of the disease_states one by one and combine identified core taxa
into a list.
list_core <- c() # an empty object to store information

for (n in group_states){ # for each variable n in group
    #print(paste0("Identifying Core Taxa for ", n))

    ps.sub <- subset_samples(Lung.group.GC376, Group == n) # Choose sample from group by n

    core_m <- core_members(ps.sub, # ps.sub is phyloseq selected with only samples from g
                           detection = 0.001, # 0.001 in atleast 90% samples
                           include.lowest = TRUE)
                           #prevalence = 0.75)
    print(paste0("No. of core taxa in ", n, " : ", length(core_m))) # print core taxa identified in
each DiseaseState.
    list_core[[n]] <- core_m # add to a list core taxa for each group.
    #print(list_core)
}

# Specify colors and plot venn
# Supplying colors in the order they appear in list_core
mycols <- c("chocolate3","green3","lightpink1")
plot(venn(list_core),
     fills = mycols)
tiff("venn_diagram_lung_GC376.tiff", units="in", width=8, height=5, res=600)
plot(venn(list_core),
     fills = mycols)
dev.off()
```

Figure 5E: Calculate Bray-Curtis distances and compare distances using boxplots in the lungs

First, make a matrix of the lung counts

```
# Filter for only lungs
remove= c("BS.244", "BS.245", "BS.246", "BS.247", "BS.248", "BS.249", "BS.250", "BS.251", "BS.252",
"BS.253", "BS.254", "BS.255", "BS.256", "BS.257", "BS.258", "BS.259", "BS.260", "BS.261", "BS.262",
"BS.263", "BS.264", "BS.265", "BS.266", "BS.267", "BS.268", "BS.269", "BS.270", "BS.271", "BS.272") #
list of rownames I would like to remove from file "data"

sample.rare.lung = sample.rare[!row.names(sample.rare)%in%remove,]
nrow(sample.rare.lung)

# Filter for only lungs with GC-376

remove= c("BS.204", "BS.213", "BS.214", "BS.215", "BS.216", "BS.217", "BS.218", "BS.219", "BS.220",
"BS.221", "BS.222", "BS.223", "BS.224", "BS.225", "BS.226", "BS.227") # list of rownames I would like
to remove from file "data"

sample.rare.lung.GC376 = sample.rare.lung[!row.names(sample.rare.lung)%in%remove,]
nrow(sample.rare.lung.GC376)
```

Next, calculate the bray-curtis distances and graph in boxplots

```
# I will calculate the bray-curtis distances
bray_lung_vehicle <- vegdist(sample.rare.lung.GC376, method = "bray", binary = FALSE)

# Turn the distance class into a matrix
bray_lung_vehicle_matrix <- as.matrix(bray_lung_vehicle)

# Export that matrix into a csv file
write.csv(bray_lung_vehicle_matrix, "bray_lung_vehicle_matrix.csv")

# Convert the matrix into a long format by hand
# I tried using a program but it omitted samples so I decided to do it by hand
bray_cecum.melt2.sd <- read.csv("/Users/bray_lung_distance.csv")

# Reorder the groups so that they are in the same order
bray_cecum.melt2.sd$Group1<- factor(bray_cecum.melt2.sd$Group1, levels = c('Mock', 'Low', 'High'))
bray_cecum.melt2.sd$Group2<- factor(bray_cecum.melt2.sd$Group2, levels = c('Mock', 'Low', 'High'))

# Plot the graph using ggplot
ggplot(bray_cecum.melt2.sd, aes(x = Group2, y = Distance)) +
  theme_bw() +
  theme(panel.grid = element_blank())+
```

```
  #geom_point() +
  geom_boxplot(aes(color = ifelse(Group1 == Group2,"goldenrod3", "black"))) +
  scale_color_identity() +
  facet_wrap(~ Group1, scales = "free_x") +
  theme(axis.text.x=element_text(hjust = 1, vjust = 0.5)) +
  scale_y_continuous(breaks = seq(0, 1, by=0.2), limits=c(0, 1))+
  ggtitle(paste0("Distance Metric = ", "Bray-Curtis")) +
  ylab("Bray-Curtis Distance") +
  xlab("Group")

# Statistics testing for comparison among means of the bray curtis for different groups (this uses the
Wilcoxon Sign test (non-parametric))
# Look at PBS group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "Mock")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)

# Look at PBS group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "Low")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)

# Look at PBS group only
bray_cecum.melt2.filter <- bray_cecum.melt2.sd %>%
  filter(Group1 == "High")
compare_means(Distance ~ Group2,  data = bray_cecum.melt2.filter)
```

Figure 5F: Visualize Bray-Curtis distances using NMDS plot in the lungs

```
# I will calculate the bray-curtis distances
# The Wisconsin transformation normalizes 0-1 so it will be very similar to the unweighted so i will
not use the transformation
sample.nmds <- metaMDS(sample.rare.lung.GC376, distance="bray", try=20, trymax=100,
autotransform=FALSE)

# Plot the bray-curtis distances in simple plot
ordiplot(sample.nmds, disp="sites")

# For more control of what your plot looks like in ggplot
sample.bray.points <- scores(sample.nmds, choices=c(1,2), disp="sites") #extract site scores for all
sites

# Duplicate the SampleID column
sample.bray.points <- cbind(rownames(sample.bray.points), data.frame(sample.bray.points,
row.names=NULL))
colnames(sample.bray.points)[1] <- "SampleID"

# Merge the bray-curtis data with the sample metadata by SampleID
sample.bray.pointsMerged <- merge(sample.bray.points, sampledataveg, by = "SampleID")

# Reorder the groups
sample.bray.pointsMerged$Group<- factor(sample.bray.pointsMerged$Group, levels = c('PBS', 'Mock-GC376',
'Infected-3-Vehicle', 'Infected-3-GC376', 'Infected-5-Vehicle', 'Infected-5-GC376'))

# Set the colors for the graph
colorgroups = c("chocolate4","forestgreen", "maroon1")

# Plot the graph using ggplot
p1 <-ggplot(sample.bray.pointsMerged, aes(x = NMDS1, y = NMDS2, fill = Group, shape = as.factor(dpi)))
+
geom_point(aes(size = 1.5))+
scale_shape_manual(values = c(21,22,23))+
stat_ellipse(aes(color = Group, group = Group), alpha = 0.3)+
scale_fill_manual(values = colorgroups)+
scale_color_manual(values = colorgroups)+
theme_bw() +
theme(panel.grid = element_blank())+
theme(strip.text.y = element_text(angle = 0))+
guides(fill=guide_legend(override.aes=list(shape=21)))+
ggsave("/Users/baseibert/Perez_Lab/Projects/Microbiome/Projects/K18_SARS/DataAnalysis/Analysis_Files/R_
Files/NMDS/NMDS_bray_lungs_GC376.png", height = 14, width = 25, units = "cm", dpi = 600)
p1

# PERMANOVA Analysis
pseq.rel <- microbiome::transform(Lung.group.GC376, "compositional")
```

```
otu <- abundances(pseq.rel)
meta <- meta(pseq.rel)

permanova <- adonis(t(otu) ~ Group,
               data = meta, permutations=1000, method = "bray")
permanova
```

Figure 5G: Visualize Bray-Curtis distances using a dendrogram in the lungs

```
# I will calculate the bray-curtis distances
bray_lung_GC376 <- vegdist(sample.rare.lung.GC376, method = "bray", binary = FALSE)

clust.res <- hclust(bray_lung_GC376, method = "average")

# Calculate cophenetic correalation coefficient
d2 <- cophenetic(clust.res)

# Compare original distance matrix with the cophenetic correalation coefficient
cor(bray_lung_GC376, d2)

# Plot the dendogram
plot(clust.res, hang = -1)
```

There are also methods for evaluating each linkage method that i will use for hclust. One method is called cophenetic correalation coefficient (CCC). It is a Pearson correlation between original distance matrix and cophenetic distances matrix of dendrogram (cluster configuration). Higher values of CCC (nearer to 1) mean better clusterization (usually values above 0.75 are considered good). Results - ward.D has a correlation of 0.78 - D2 has a correlation of 0.78 - single has a correlation of 0.75 - complete has a correlation of 0.78 - average (UPGMA) has a correlation of 0.79 - mcquitty (WPGMA) has a correlation of 0.73 - median (WPGMC) has a correlation of 0.69 - centroid (UPGMC) has a correlation of 0.65 **I will use UPGMA or average since it has the highest correlation**

In the dendrogram, the y-axis is simply the value of this distance metric between clusters. For example, if you see two clusters merged at a height x, it means that the distance between those clusters was x

The colors of groups and dpc were added in illustrator.