Sealice microbiome

Bela Klimesova

2025-12-12

Contents

1 Loading microbiome datasets

```
library(phyloseq)
library(ggplot2)
suppressMessages(library(vegan))
suppressMessages(library(microbiomeMarker))
library(agricolae)
suppressMessages(library(tidyverse))
library(patchwork)
suppressMessages(library(ggvenn))
```

Data loading: Microbiome read count data

```
Sequen<-read.table("Sealice_Microbiome_Reads.csv",
    header = TRUE,sep = ',',fill = TRUE,dec = ".",na.strings = "NA")
```

Adjustment of dataset and devision to Illumina and Nanopore data

```
rownames(Sequen)<-Sequen[,1]
Sequen[,1]<- NULL

Sequen_t<-t(Sequen)

Illumina<-Sequen[,c(1:20)]
Nanopore<-Sequen[,c(21:40)]
```

Data loading: Microbiome metadata

```
sample<-read.csv("Sealice_Microbiome_Metadata.txt",header=T,row.names=1,sep = "\t")
```

Adjustment of dataset and devision to Illumina and Nanopore data

```
sample$time<-rep(c(rep('February',5),rep('March',5),rep('April',5),rep('May',5)),2)
sample$platform<-c(rep('Illumina',20),rep('Nanopore',20))

Ill_sample<-sample[sample$platform=='Illumina',]
Nan_sample<-sample[sample$platform=='Nanopore',]
```

Data loading: Microbiome taxonomy data

```
observation<-read.csv("Sealice_Microbiome_Taxonomy.tsv",header=T,row.names=1, sep = "\t")
```

2 Dataset format adjustment

Discarding global singletons in all three datasets (Illumina, Nanopore, Overall)

```
Singletons<-names(apply(Illumina,1,sum)[apply(Illumina,1,sum)<2])
Illumina_1<-Illumina[!(rownames(Illumina)%in% Singletons),]

Singletons<-names(apply(Nanopore,1,sum)[apply(Nanopore,1,sum)<2])
Nanopore_1<-Nanopore[!(rownames(Nanopore)%in% Singletons),]

Singletons<-names(apply(Sequen,1,sum)[apply(Sequen,1,sum)<2])
Sequen_1<-Sequen[!(rownames(Sequen)%in% Singletons),]
```

Data to phyloseq format

Illumina

```
OTU1 = otu_table(as.matrix(t(Illumina_1)), taxa_are_rows = FALSE)
SAM1 = sample_data(Ill_sample, errorIfNULL = TRUE)
TAX1 = tax_table(as.matrix(observation))
Ill_phylo <- phyloseq(OTU1, TAX1, SAM1)
```

Nanopore

```
OTU2 = otu_table(as.matrix(t(Nanopore_1)), taxa_are_rows = FALSE)
SAM2 = sample_data(Nan_sample, errorIfNULL = TRUE)
TAX2 = tax_table(as.matrix(observation))
Nan_phylo <- phyloseq(OTU2, TAX2, SAM2)
```

Overall

```
OTU = otu_table(as.matrix(t(Sequen)), taxa_are_rows = FALSE)
SAM = sample_data(sample, errorIfNULL = TRUE)
TAX = tax_table(as.matrix(observation))
phylo <- phyloseq(OTU, TAX, SAM)
```

3 Data analysis

4 Rarefaction curves

Adjustment of data fortmat

```
Illumina_t <- t(Illumina_1)
Nanopore_t <- t(Nanopore_1)
```

Names of samples and matching colors

```
leg.txt <- c(paste0(1:5,'_Feb'),paste0(1:5,'_Mar'),
        paste0(1:5,'_Apr'),paste0(1:5,'_May'))
col_vector <- c("#661100","#7F261D","#993B3B","#B25059","#CC6677",
            "#DDCC77","#AAB666","#77A155","#438C44","#117733",
            "#88CCEE","#72A1D4","#5D77BB","#484CA1","#332288",
            "#44AA99","#55A194","#669990","#77908C","#888888")
```

Figure of rarefaction curves

```
layout(matrix(c(1,2,3,3), nrow = 2, byrow = TRUE),
       heights = c(3, 1))  # top row bigger than bottom

par(mar = c(5, 4, 4, 2), cex = 1, las = 1)

# Illumina
rarecurve(Illumina_t,
          step = 100, lwd = 2, cex = 0.8,
          xlab = "Number of reads",
          ylab = "Number of OTU observed",
          ylim = c(1, 350), xlim = c(1, 250000),
          col = col_vector, size = 1.5,
          label = FALSE, main = "A) Illumina")

# Nanopore
rarecurve(Nanopore_t,
          step = 100, lwd = 2, cex = 0.8,
          xlab = "Number of reads",
          ylab = "Number of OTU observed",
          ylim = c(1, 350), xlim = c(1, 250000),
          col = col_vector, size = 1.5,
          label = FALSE, main = "B) Nanopore")

par(mar = c(0, 0, 0, 0))
plot.new()
legend("center",
       legend = leg.txt, lwd = 2, col = col_vector,
       box.lwd = 0.6, cex = 1.0, ncol = 5)
```

Determination of sequencing depth

Illumina

```
sum_seq <- colSums(Illumina)
range(sum_seq)
```

```
## [1]  44100 115724
```

```
sum(sum_seq)
```

```
## [1] 1622493
```

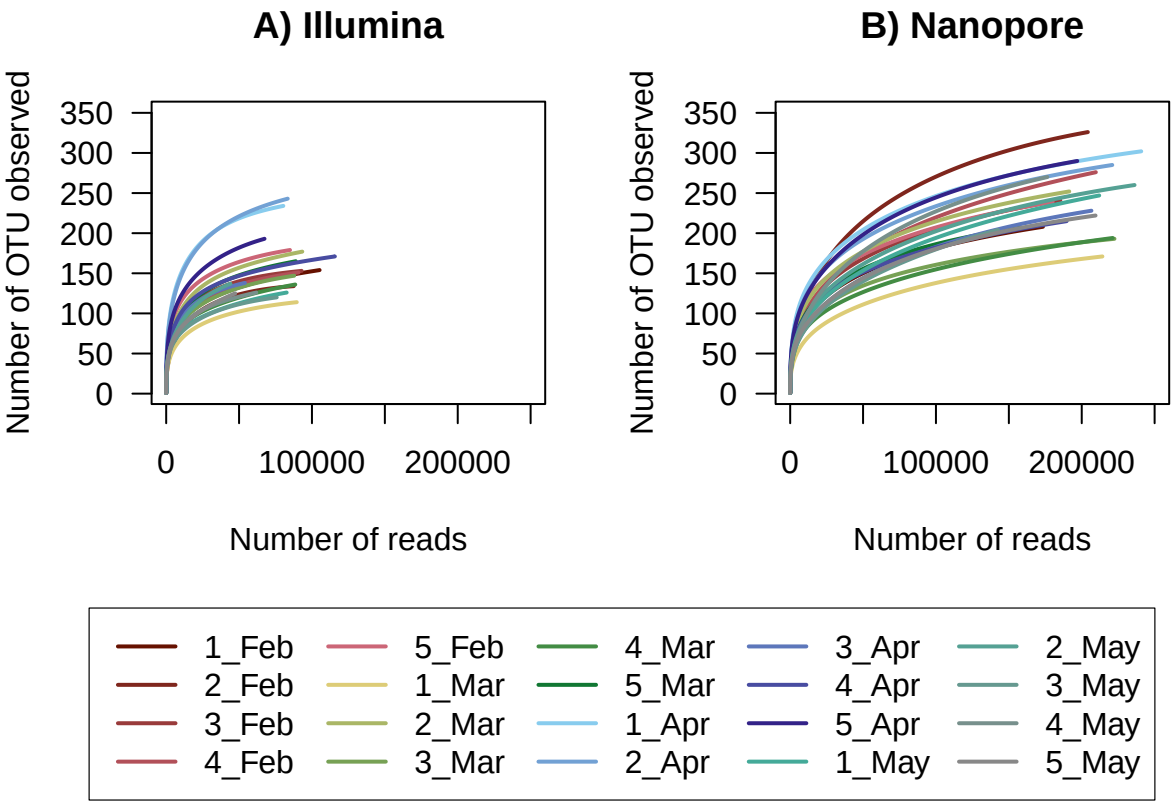

Figure 1: Rarefaction curve

```
mean(sum_seq)
```

```
## [1] 81124.65
```

```
sd(sum_seq)
```

```
## [1] 18196.91
```

Nanopore

```
sum_seq <- colSums(Nanopore)
range(sum_seq)
```

```
## [1] 142570 240831
```

```
sum(sum_seq)
```

```
## [1] 3991554
```

```
mean(sum_seq)
```

```
## [1] 199577.7
```

```
sd(sum_seq)
```

```
## [1] 25252.9
```

5 Alpha diversity and richness

Calculation of alpha diversity

Illumina

```
data_richness <- estimateR(Illumina_t)
data_evenness <- diversity(Illumina_t) / log(specnumber(Illumina_t))
data_shannon <- diversity(Illumina_t, index = "shannon")
data_simpson <- diversity(Illumina_t, index = "simpson")
data_alphadiv1 <- cbind(Ill_sample, t(data_richness), data_shannon,
                        data_simpson, data_evenness)
```

Nanopore

```
data_richness <- estimateR(Nanopore_t)
data_evenness <- diversity(Nanopore_t) / log(specnumber(Nanopore_t))
data_shannon <- diversity(Nanopore_t, index = "shannon")
data_simpson <- diversity(Nanopore_t, index = "simpson")
data_alphadiv2 <- cbind(Nan_sample, t(data_richness), data_shannon,
                        data_simpson, data_evenness)
```

Combining dataset

```
data_alphadiv<-rbind(as.data.frame(data_alphadiv1),as.data.frame(data_alphadiv2))
data_alphadiv$time<-factor(data_alphadiv$time,levels=c('February','March','April','May'))
```

5.1 Shannon index

Test normal distribution

```
shapiro.test(data_alphadiv1$data_shannon)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv1$data_shannon
## W = 0.94404, p-value = 0.2855
```

```
shapiro.test(data_alphadiv2$data_shannon)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv2$data_shannon
## W = 0.96005, p-value = 0.5449
```

ANOVA test to assess differences

```
aov_test1<-aov(data_shannon ~ time, data = data_alphadiv1)
summary(aov_test1)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## time         3  2.027  0.6758   9.842 0.000644 ***
## Residuals   16  1.099  0.0687
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov_test2<-aov(data_shannon ~ time, data = data_alphadiv2)
summary(aov_test2)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## time         3  2.140  0.7133   8.046 0.00171 **
## Residuals   16  1.418  0.0887
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Post-hoc Tukey test

```
hsd_test1 <- TukeyHSD(aov_test1)
hsd_res1 <- HSD.test(aov_test1, "time", group=T)$groups
hsd_res1
```

```
##          data_shannon groups
## April        2.518339      a
## May          2.090393     ab
## February     1.954002      b
## March        1.631316      b
```

```
hsd_test2 <- TukeyHSD(aov_test2)
hsd_res2 <- HSD.test(aov_test2, "time", group=T)$groups
hsd_res2
```

```
##          data_shannon groups
## April        2.116076      a
## May          1.877787     ab
## February     1.560386     bc
## March        1.248588      c
```

Preparation of data for plotting

```
summary_data <- data_alphadiv %>%
  group_by(time, platform) %>%
  summarise(mean = mean(data_shannon, na.rm = TRUE),
            sd = sd(data_shannon, na.rm = TRUE),
            .groups = 'drop')

label_data <- summary_data %>%
  mutate(label = c("a", "ab", "a", "b", "b", "c", "ab", "bc"),
         y = mean + sd + 0.15)
```

Figure of Shannon index

```
P1 <- ggplot(data_alphadiv, aes(x = time, y = data_shannon, fill = time,alpha=0.5)) +
    geom_errorbar(data = summary_data,
                aes(x = time, ymin = mean - sd, ymax = mean + sd, group = platform),
                position = position_dodge(width = 0.75),
                width = 0.2,
                color = "black",
                inherit.aes = FALSE)+
    geom_text(data = label_data,
            aes(x = time, y = y, label = label,group = platform),
            position = position_dodge(width = 0.75),
            inherit.aes = FALSE,
            size = 4) +
    geom_boxplot(position = position_dodge(width = 0.75), coef = 0,outlier.shape = NA) +
    scale_fill_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
    scale_color_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
    labs(title = 'A)  Shannon index', x= ' ', y= '') +
    theme_minimal()+
    facet_wrap(~platform)+
    theme(legend.position="none",
    axis.text.x = element_text(angle = 90, hjust = 1))
```

5.2 Simpson index

Test normal distribution

```
shapiro.test(data_alphadiv1$data_simpson)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv1$data_simpson
## W = 0.95181, p-value = 0.3955
```

```
shapiro.test(data_alphadiv2$data_simpson)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv2$data_simpson
## W = 0.93885, p-value = 0.228
```

ANOVA test to assess differences

```
aov_test1<-aov(data_simpson ~ time, data = data_alphadiv1)
summary(aov_test1)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## time         3 0.1442 0.04808   6.538 0.00429 **
## Residuals   16 0.1177 0.00735
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov_test2<-aov(data_simpson ~ time, data = data_alphadiv2)
summary(aov_test2)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## time         3 0.2387 0.07957   6.782 0.00367 **
## Residuals   16 0.1877 0.01173
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Post-hoc Tukey test

```
hsd_test1 <- TukeyHSD(aov_test1)
hsd_res1 <- HSD.test(aov_test1, "time", group=T)$groups
hsd_res1
```

```
##          data_simpson groups
## April       0.7904461      a
## May         0.7203377      a
## February    0.6638329     ab
## March       0.5583288      b
```

```
hsd_test2 <- TukeyHSD(aov_test2)
hsd_res2 <- HSD.test(aov_test2, "time", group=T)$groups
hsd_res2
```

```
##          data_simpson groups
## April       0.6901969      a
## May         0.6302614      a
## February    0.5153539     ab
## March       0.4055207      b
```

Preparation of data for plotting

```
summary_data <- data_alphadiv %>%
  group_by(time, platform) %>%
  summarise(mean = mean(data_simpson, na.rm = TRUE),
            sd = sd(data_simpson, na.rm = TRUE),
            .groups = 'drop')

label_data <- summary_data %>%
  mutate(label = c("ab", "ab", "a", "a", "b", "b", "b", "b"),
         y = mean + sd + 0.05)
```

Figure of Simpson index

```
P2 <- ggplot(data_alphadiv, aes(x = time, y = data_simpson, fill = time,alpha=0.5)) +
  geom_errorbar(data = summary_data,
                aes(x = time, ymin = mean - sd, ymax = mean + sd, group = platform),
                position = position_dodge(width = 0.75),
                width = 0.2,
                color = "black",
                inherit.aes = FALSE)+
  geom_text(data = label_data,
            aes(x = time, y = y, label = label,group = platform),
            position = position_dodge(width = 0.75),
            inherit.aes = FALSE,
            size = 4) +
  geom_boxplot(position = position_dodge(width = 0.75), coef = 0,outlier.shape = NA) +
    scale_fill_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
    scale_color_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
  labs(title = 'B)  Simpson index', x= ' ', y= '') +
  theme_minimal()+
    facet_wrap(~platform)+
    theme(legend.position="none",
    axis.text.x = element_text(angle = 90, hjust = 1))
```

5.3 Observed richness

Test normal distribution

```r
shapiro.test(data_alphadiv1$S.obs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv1$S.obs
## W = 0.87777, p-value = 0.01614
```

```r
shapiro.test(data_alphadiv2$S.obs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv2$S.obs
## W = 0.97947, p-value = 0.9271
```

ANOVA and Kruskal test to assess differences

```r
Krus<-kruskal.test(S.obs ~ time, data = data_alphadiv1)
pairwise.wilcox.test(data_alphadiv1$S.obs,data_alphadiv1$time,
                 p.adjust.method = "BH")
```

```
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
##
## data:  data_alphadiv1$S.obs and data_alphadiv1$time
##
##          April February March
## February 0.226 -        -
## March    0.190 0.690    -
## May      0.064 0.064    0.250
##
## P value adjustment method: BH
```

```r
aov_test2<-aov(S.obs ~ time, data = data_alphadiv2)
summary(aov_test2)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## time         3  11155    3718    2.84 0.0709 .
## Residuals   16  20944    1309
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Preparation of data for plotting

```r
summary_data <- data_alphadiv %>%
  group_by(time, platform) %>%
  summarise(mean = mean(S.obs, na.rm = TRUE),
            sd = sd(S.obs, na.rm = TRUE),
            .groups = 'drop')
```

Figure of observed richness

```
P3 <- ggplot(data_alphadiv, aes(x = time, y = S.obs, fill = time,alpha=0.5)) +
  geom_errorbar(data = summary_data,
                aes(x = time, ymin = mean - sd, ymax = mean + sd, group = platform),
                position = position_dodge(width = 0.75),
                width = 0.2,
                color = "black",
                inherit.aes = FALSE) +
  geom_boxplot(position = position_dodge(width = 0.75), coef = 0,outlier.shape = NA) +
    scale_fill_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
    scale_color_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
  labs(title = 'C)   Observed richness', x = '', y = '') +
  theme_minimal()+
    facet_wrap(~platform)+
    theme(legend.position="none",
    axis.text.x = element_text(angle = 90, hjust = 1))
```

5.4 ACE

Test normal distribution

```
shapiro.test(data_alphadiv1$S.ACE)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv1$S.ACE
## W = 0.93012, p-value = 0.1552
```

```
shapiro.test(data_alphadiv2$S.ACE)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data_alphadiv2$S.ACE
## W = 0.97571, p-value = 0.8676
```

ANOVA test to assess differences

```
aov_test1<-aov(S.ACE ~ time, data = data_alphadiv1)
summary(aov_test1)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## time         3  11224    3741   3.964 0.0274 *
## Residuals   16  15102     944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov_test2<-aov(S.ACE ~ time, data = data_alphadiv2)
summary(aov_test2)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## time         3  15992    5331    3.62 0.0362 *
## Residuals   16  23561    1473
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Post-hoc Tukey test

```r
hsd_test1 <- TukeyHSD(aov_test1)
hsd_res1 <- HSD.test(aov_test1, "time", group=T)$groups
hsd_res1
```

```
##             S.ACE groups
## April    218.7627      a
## March    170.5121     ab
## February 169.4364     ab
## May      156.6333      b
```

```r
hsd_test2 <- TukeyHSD(aov_test2)
hsd_res2 <- HSD.test(aov_test2, "time", group=T)$groups
hsd_res2
```

```
##             S.ACE groups
## February 321.8522      a
## April    321.2657      a
## May      315.0956      a
## March    254.3877      a
```

Preparation of data for plotting

```r
summary_data <- data_alphadiv %>%
  group_by(time, platform) %>%
  summarise(mean = mean(S.ACE, na.rm = TRUE),
            sd = sd(S.ACE, na.rm = TRUE),
            .groups = 'drop')

label_data <- summary_data %>%
  mutate(label = c("ab", "", "ab", "", "a", "", "b", ""),
         y = mean + sd + 40)
```

Figure of ACE

```r
P4 <- ggplot(data_alphadiv, aes(x = time, y = S.ACE, fill = time,alpha=0.5)) +
  geom_errorbar(data = summary_data,
                aes(x = time, ymin = mean - sd, ymax = mean + sd, group = platform),
                position = position_dodge(width = 0.75),
                width = 0.2,
```

```
                color = "black",
                inherit.aes = FALSE) +
  geom_text(data = label_data,
            aes(x = time, y = y, label = label,group = time),
            position = position_dodge(width = 0.75),
            inherit.aes = FALSE,
            size = 4) +
  geom_boxplot(position = position_dodge(width = 0.75), coef = 0,outlier.shape = NA) +
    scale_fill_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
    scale_color_manual(values = c("#CC6677","#DDCC77","#117733","#332288")) +
  labs(title = 'D)   ACE', x = '', y = '') +
  theme_minimal()+
  facet_wrap(~platform)+
    theme(legend.position="none",
    axis.text.x = element_text(angle = 90, hjust = 1))
```

Combined plot of richness and diversity

```
((P1 | P2)/(P3 | P4))
```

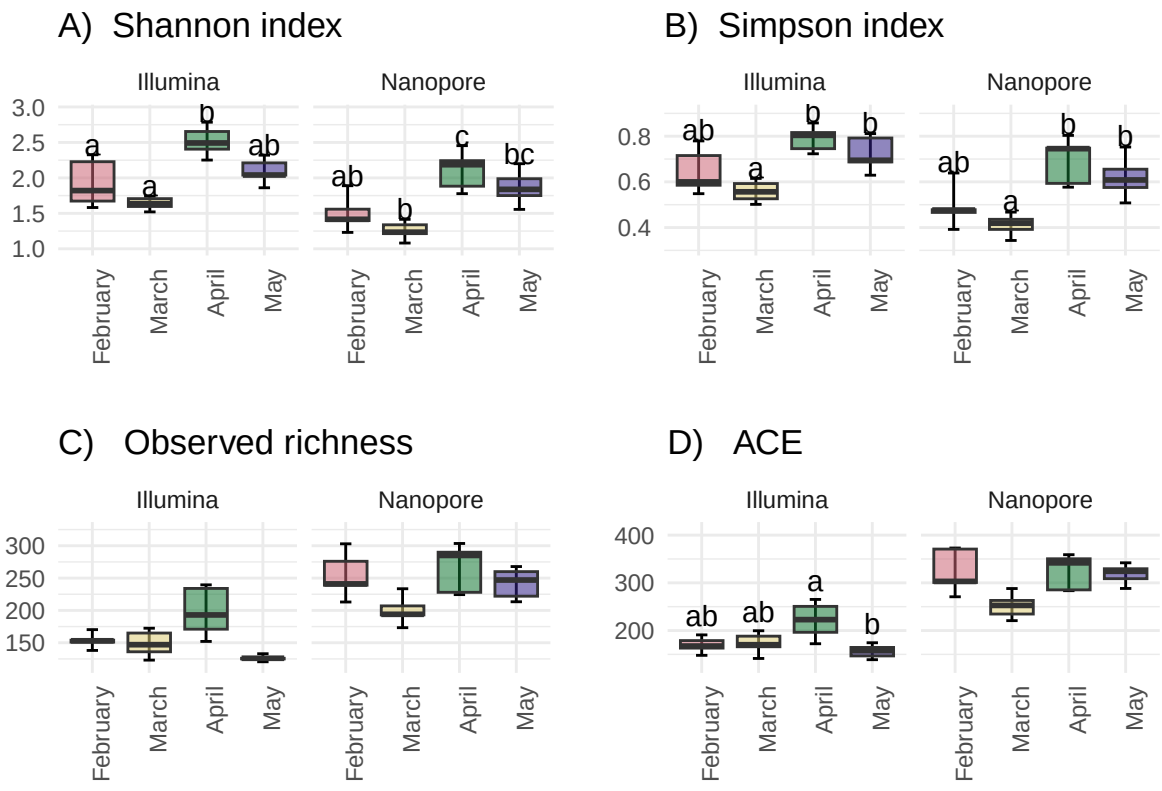

Figure 2: Diversity and richness

6 Data normalization = calculation of abundance

```
Ill_phylo_norm <- transform_sample_counts(Ill_phylo, function(ASV) ASV/sum(ASV) *100 )
Nan_phylo_norm <- transform_sample_counts(Nan_phylo, function(ASV) ASV/sum(ASV) *100 )
phylo_norm <- transform_sample_counts(phylo, function(ASV) ASV/sum(ASV) *100 )
```

7 Data filtering

Keep only taxa (columns) where at least one sample has 0.03 abundance

```
threshold <- 0.03
```

Illumina

```
Ill_atu_filt <- otu_table(Ill_phylo_norm)[, colSums(otu_table(Ill_phylo_norm)
                                                    >= threshold) > 0]
ncol(Ill_atu_filt)
```

```
## [1] 132
```

```
ncol(otu_table(Ill_phylo_norm))
```

```
## [1] 482
```

Nanopore

```
Nan_atu_filt <- otu_table(Nan_phylo_norm)[, colSums(otu_table(Nan_phylo_norm)
                                                    >= threshold) > 0]
ncol(Nan_atu_filt)
```

```
## [1] 134
```

```
ncol(otu_table(Nan_phylo_norm))
```

```
## [1] 726
```

Overall

```
atu_filt <- otu_table(phylo_norm)[, colSums(otu_table(phylo_norm)
                                            >= threshold) > 0]
ncol(atu_filt)
```

```
## [1] 160
```

```
ncol(otu_table(phylo_norm))
```

```
## [1] 1127
```

Overview of detected taxonomical groups Illumina

```
Names<-colnames(Ill_atu_filt)
Selec<-data.frame(TAX1[rownames(TAX1) %in% Names,])

length(unique(Selec$Phylum))
```

```
## [1] 11
```

```
length(unique(Selec$Class))
```

```
## [1] 13
```

```
length(unique(Selec$Order))
```

```
## [1] 35
```

Nanopore

```
Names<-colnames(Nan_atu_filt)
Selec<-data.frame(TAX2[rownames(TAX2) %in% Names,])

length(unique(Selec$Phylum))
```

```
## [1] 13
```

```
length(unique(Selec$Class))
```

```
## [1] 17
```

```
length(unique(Selec$Order))
```

```
## [1] 39
```

8 Beta diversity

```
set.seed(1782)
```

Changing the format of data

```
SAM1$grouping<-c(rep('Feb and Mar',10),rep('Apr and May',10))
Ill_phylo_filt_1 <- phyloseq(otu_table(Ill_atu_filt), TAX1, SAM1)

SAM2$grouping<-c(rep('Feb and Mar',10),rep('Apr and May',10))
Nan_phylo_filt_1 <- phyloseq(otu_table(Nan_atu_filt), TAX2, SAM2)
phylo_filt_1 <- phyloseq(otu_table(atu_filt), TAX, SAM)
```

Principal coordinate analysis

Illumina

```
pcoa_bc1 = ordinate(Ill_phylo_filt_1, "PCoA", "bray")
P1<-plot_ordination(Ill_phylo_filt_1, pcoa_bc1, color = "time") +
    geom_point(size = 3,alpha=0.5) +
    theme_minimal()+
    scale_color_manual(values=c("#CC6677","#DDCC77","#117733","#332288"),
        breaks = c('February','March','April','May'))+
    labs(title = 'A)   Illumina', x= 'Component [40.6%]',
        y= 'Component [20.8%]') +
    theme(text = element_text(size = 13))+
  stat_ellipse(aes(group = grouping), linetype = 2,alpha=0.5)
```

Nanopore

```
pcoa_bc2 = ordinate(Nan_phylo_filt_1, "PCoA", "bray")
P2<-plot_ordination(Nan_phylo_filt_1, pcoa_bc2, color = "time") +
    geom_point(size = 3,alpha=0.5) +
    theme_minimal()+
    scale_color_manual(values=c("#CC6677","#DDCC77","#117733","#332288"),
        breaks = c('February','March','April','May'))+
    labs(title = 'B)   Nanopore', x= 'Component [51.3%]',
        y= 'Component [22.7%]')+
    theme(text = element_text(size = 13)) +
  stat_ellipse(aes(group = grouping), linetype = 2,alpha=0.5)
```

Combining of figures for beta diversity

```
((P1 | P2 ))  +
  plot_layout(guides = "collect") &
  theme(legend.position = "bottom")
```

Testing differences between months with PERMANOVA Illumina

```
adonis2(Ill_atu_filt~time,data=Ill_sample, permutations=9999, method="bray")
```

```
## Permutation test for adonis under reduced model
## Permutation: free
## Number of permutations: 9999
##
## adonis2(formula = Ill_atu_filt ~ time, data = Ill_sample, permutations = 9999, method = "bray")
##          Df SumOfSqs      R2      F Pr(>F)
```

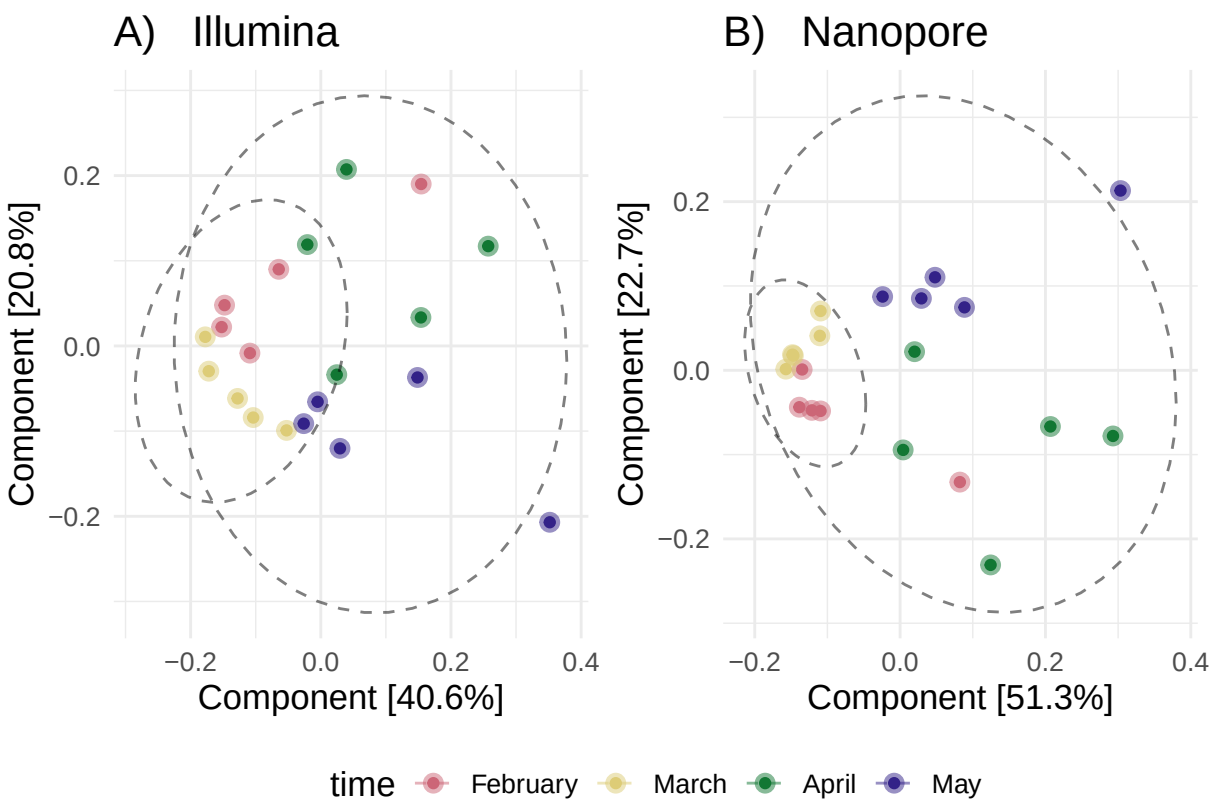

Figure 3: Beta-diversity: PCoA

```
## Model     3  0.42751 0.41792 3.8293  1e-04 ***
## Residual 16  0.59543 0.58208
## Total    19  1.02294 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nanopore

```
adonis2(Nan_atu_filt~time,data=Nan_sample, permutations=9999, method="bray")
```

```
## Permutation test for adonis under reduced model
## Permutation: free
## Number of permutations: 9999
##
## adonis2(formula = Nan_atu_filt ~ time, data = Nan_sample, permutations = 9999, method = "bray")
##          Df SumOfSqs     R2      F Pr(>F)
## Model     3  0.43973 0.5469 6.4374  1e-04 ***
## Residual 16  0.36432 0.4531
## Total    19  0.80405 1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

9 Microbiome composition: phylum level

Adjusting format of data

```
observation$SampleID<-rownames(observation)
```

Illumina

```
Ill_atu_filt_rar<-data.frame(t(Ill_atu_filt))
Ill_atu_filt_rar$SampleID<-rownames(Ill_atu_filt_rar)
```

Nanopore

```
Nan_atu_filt_rar<-data.frame(t(Nan_atu_filt))
Nan_atu_filt_rar$SampleID<-rownames(Nan_atu_filt_rar)
```

Overall

```
atu_filt_rar<-data.frame(t(atu_filt))
atu_filt_rar$SampleID<-rownames(atu_filt_rar)
```

Merging datasets

```
Tax_Data1<-merge(Ill_atu_filt_rar,observation,all.x=TRUE,by='SampleID')
colnames(Tax_Data1)[2:21]<-paste0(rep(c(1:5),4),'_',sample$time[1:20])

Tax_Data2<-merge(Nan_atu_filt_rar,observation,all.x=TRUE,by='SampleID')
colnames(Tax_Data2)[2:21]<-paste0(rep(c(1:5),4),'_',sample$time[1:20])

Tax_Data<-merge(atu_filt_rar,observation,all.x=TRUE,by='SampleID')
```

Sum per Phyla

```
Tax_Phylum1<- aggregate(Tax_Data1[,2:21],by=list(Phylum=Tax_Data1$Phylum), sum)
Tax_Phylum1$Phylum<-substr(Tax_Phylum1$Phylum,4,40)

Tax_Phylum2<- aggregate(Tax_Data2[,2:21],by=list(Phylum=Tax_Data2$Phylum), sum)
Tax_Phylum2$Phylum<-substr(Tax_Phylum2$Phylum,4,40)
```

Tidying datasets Illumina

```
Tax_Phylum1<-data.frame(t(Tax_Phylum1))
colnames(Tax_Phylum1)<-Tax_Phylum1[1,]
Tax_Phylum1<-Tax_Phylum1[2:21,]
Tax_Phylum1$month<-substr(rownames(Tax_Phylum1),3,10)
Tax_Phylum1$Sample_ID<-rownames(Tax_Phylum1)
Tax_Phylum1$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Phyll<- gather(Tax_Phylum1,Phylums,Abundance,
                   Actinomycetota:Verrucomicrobiota, factor_key=TRUE)
Tax_Phyll$Abundance_r<-round(as.numeric(Tax_Phyll$Abundance),2)
```

Nanopore

```
Tax_Phylum2<-data.frame(t(Tax_Phylum2))
colnames(Tax_Phylum2)<-Tax_Phylum2[1,]
Tax_Phylum2<-Tax_Phylum2[2:21,]
Tax_Phylum2$month<-substr(rownames(Tax_Phylum2),3,10)
Tax_Phylum2$Sample_ID<-rownames(Tax_Phylum2)
Tax_Phylum2$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Phyl2<- gather(Tax_Phylum2,Phylums,Abundance,
                   Bacillota:Verrucomicrobiota, factor_key=TRUE)
Tax_Phyl2$Abundance_r<-round(as.numeric(Tax_Phyl2$Abundance),2)
```

Filtering: phyla with abundance lower than 0.5% described as others

```
Tax_Phyll$Phylum<-as.vector(Tax_Phyll$Phylums)
Tax_Phyll[Tax_Phyll$Abundance_r<0.5,]$Phylum<-'Others'

Tax_Phyl2$Phylum<-as.vector(Tax_Phyl2$Phylums)
Tax_Phyl2[Tax_Phyl2$Abundance_r<0.5,]$Phylum<-'Others'
```

Combining datasets

```
Tax_Phyll$platform<-'Illumina'
Tax_Phyl2$platform<-'Nanopore'
Tax_Phyl<-rbind(Tax_Phyll,Tax_Phyl2)
```

Adjustment of final dataset

```
Tax_Phyl_N<-aggregate(Tax_Phyl$Abundance_r,list(month=Tax_Phyl$month,
            Phylum=Tax_Phyl$Phylum,
            platform=Tax_Phyl$platform),mean)
Tax_Phyl_N$month<-factor(Tax_Phyl_N$month,levels = c('February','March','April','May'))

sort(tapply(Tax_Phyl_N$x,Tax_Phyl_N$Phylum,sum))
```

```
##             Myxococcota                  Others               Bacillota
##                0.510000                0.532574                0.910000
## Thermodesulfobacteriota        Bdellovibrionota         Cyanobacteriota
##                1.700000                2.437500                6.666000
##         Patescibacteria        Campylobacterota       Verrucomicrobiota
##                11.826500               20.663000               25.308000
##             Bacteroidota          Pseudomonadota
##               103.698000              634.768000
```

```
Tax_Phyl_N$Phylum<-factor(Tax_Phyl_N$Phylum,
    levels = c("Myxococcota",'Others','Bacillota',"Thermodesulfobacteriota",
    "Bdellovibrionota","Cyanobacteriota",'Patescibacteria',"Campylobacterota",
    "Verrucomicrobiota","Bacteroidota","Pseudomonadota"))
```

Figure of bacterial compositon at phylum level

```
P_Phyl<-ggplot(Tax_Phyl_N,aes(x = month, y = x, fill = Phylum)) +
  geom_bar(position = "fill", stat = "identity") +
  scale_fill_manual(values = c("#888888","#44AA99","#882255","#6699CC",
    "#999933","#AA4499","#88CCEE","#332288","#117733","#CC6677","#DDCC77")) +
  labs(title= 'A)', x= 'Month',
    y= 'Proportion of the total abundance')+
    facet_wrap(~platform) +
  theme_minimal()+
    theme(text = element_text(size = 17),
    axis.text.x = element_text(angle = 90, hjust = 1))
```

```
(P_Phyl)
```

Range of abundance values Illumina

```
tapply(Tax_Phyll$Abundance_r,Tax_Phyll$Phylum,mean)
```

```
##             Bacteroidota        Bdellovibrionota        Campylobacterota
##              16.65950000              0.70800000              1.63416667
##         Cyanobacteriota                  Others         Patescibacteria
##              1.26750000              0.06913043              1.41947368
##          Pseudomonadota Thermodesulfobacteriota       Verrucomicrobiota
##              76.35000000              1.14000000              3.21100000
```

```
range(Tax_Phyll[Tax_Phyll$Phylum=='Pseudomonadota',]$Abundance_r)
```

```
## [1] 61.66 87.57
```

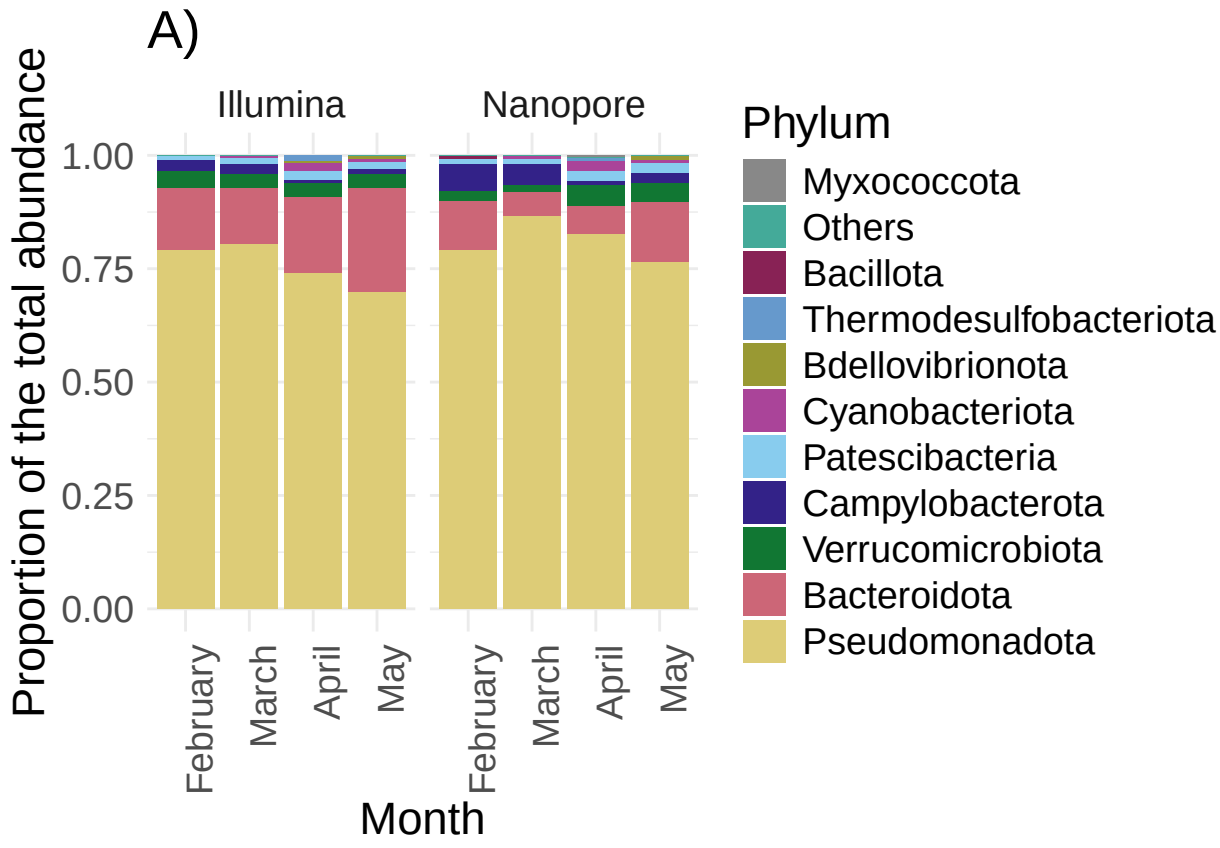

Figure 4: Microbiome composition: phylum level

```
range(Tax_Phyll[Tax_Phyll$Phylum=='Bacteroidota',]$Abundance_r)
```

```
## [1]  6.33 27.56
```

Nanopore

```
tapply(Tax_Phyl2$Abundance_r,Tax_Phyl2$Phylum,mean)
```

```
##               Bacillota            Bacteroidota        Bdellovibrionota
##              0.91000000              9.26500000              1.11000000
##        Campylobacterota         Cyanobacteriota             Myxococcota
##              3.25923077              1.71750000              0.51000000
##                  Others         Patescibacteria          Pseudomonadota
##              0.06602564              1.59526316             82.34200000
## Thermodesulfobacteriota       Verrucomicrobiota
##              0.56000000              3.11600000
```

```
range(Tax_Phyl2[Tax_Phyl2$Phylum=='Pseudomonadota',]$Abundance_r)
```

```
## [1] 66.63 91.49
```

```
range(Tax_Phyl2[Tax_Phyl2$Phylum=='Bacteroidota',]$Abundance_r)
```

```
## [1]  3.45 19.88
```

10 Bacterial composition: class level

Adjusting the dataset to class level Illumina

```
Tax_Data1[Tax_Data1$Class=="c__Incertae_Sedis",]$Class<-'p__NB1_j_Inc_Sed'

Tax_Class1<- aggregate(Tax_Data1[,2:21],by=list(Class=Tax_Data1$Class), sum)
Tax_Class1$Class<-substr(Tax_Class1$Class,4,40)

Tax_Class1<-data.frame(t(Tax_Class1))
colnames(Tax_Class1)<-Tax_Class1[1,]
Tax_Class1<-Tax_Class1[2:21,]
Tax_Class1$month<-substr(rownames(Tax_Class1),3,10)
Tax_Class1$Sample_ID<-rownames(Tax_Class1)
Tax_Class1$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Cla1<- gather(Tax_Class1,Class,
                  Abundance,Acidimicrobiia:NB1_j_Inc_Sed, factor_key=TRUE)
Tax_Cla1$Abundance_r<-round(as.numeric(Tax_Cla1$Abundance),4)
```

Nanopore

```
Tax_Data2[Tax_Data2$Class=="c__Incertae_Sedis",]$Class<-'p__Marinimicrobia_inc._sed.'

Tax_Class2<- aggregate(Tax_Data2[,2:21],by=list(Class=Tax_Data2$Class), sum)
Tax_Class2$Class<-substr(Tax_Class2$Class,4,40)

Tax_Class2<-data.frame(t(Tax_Class2))
colnames(Tax_Class2)<-Tax_Class2[1,]
Tax_Class2<-Tax_Class2[2:21,]
Tax_Class2$month<-substr(rownames(Tax_Class2),3,10)
Tax_Class2$Sample_ID<-rownames(Tax_Class2)
Tax_Class2$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Cla2<- gather(Tax_Class2,Class,Abundance,
                  Alphaproteobacteria:Marinimicrobia_inc._sed., factor_key=TRUE)
Tax_Cla2$Abundance_r<-round(as.numeric(Tax_Cla2$Abundance),4)
```

Filtering only 10 orders with the highest abundance

```
Part<-names(sort(tapply(as.numeric(Tax_Cla1$Abundance),Tax_Cla1$Class,mean)))[4:13]
Tax_Cla1$Class_N<-as.character(Tax_Cla1$Class)
Tax_Cla1[!(Tax_Cla1$Class_N %in% Part),]$Class_N<-'Others'
Tax_Cla1$platform<-'Illumina'

Part<-names(sort(tapply(as.numeric(Tax_Cla2$Abundance),Tax_Cla2$Class,mean)))[8:17]
Tax_Cla2$Class_N<-as.character(Tax_Cla2$Class)
Tax_Cla2[!(Tax_Cla2$Class_N %in% Part),]$Class_N<-'Others'
Tax_Cla2$platform<-'Nanopore'
```

Combining datasets

```
Tax_Cla<-rbind(Tax_Cla1,Tax_Cla2)

Tax_Cla_New<-aggregate(Tax_Cla$Abundance_r,list(month=Tax_Cla$month,
            Class=Tax_Cla$Class_N,platform=Tax_Cla$platform),mean)

Tax_Cla_New$month<-factor(Tax_Cla_New$month,levels=c('February','March','April','May'))
Tax_Cla_New$Class<-factor(Tax_Cla_New$Class,
    levels = c('Others','Desulfuromonadia','Polyangiia','Bacteriovoracia',
        'Cyanobacteriia','Gracilibacteria','Campylobacteria',
        'Verrucomicrobiia','Alphaproteobacteria','Bacteroidia',
        'Gammaproteobacteria'))
sort(tapply(Tax_Cla_New$x,Tax_Cla_New$Class,mean))
```

```
##              Others    Desulfuromonadia          Polyangiia     Bacteriovoracia
##          0.01637774          0.07577750          0.10838500          0.20178750
##      Cyanobacteriia     Gracilibacteria     Campylobacteria    Verrucomicrobiia
##          0.68004750          1.44870750          1.64051750          3.16018750
## Alphaproteobacteria         Bacteroidia Gammaproteobacteria
##          9.22959500         12.96208500         70.11614000
```

Figure of bacterial compositon at class level

```
P_Cla<-ggplot(Tax_Cla_New,aes(x = month, y = x, fill = Class)) +
  geom_bar(position = "fill", stat = "identity") +
  scale_fill_manual(values = c("#332288","#117733","#CC6677","#DDCC77","#888888",
        "#6699CC","#882255","#999933","#44AA99","#AA4499","#88CCEE"
        )) +
  labs(title= 'C)', x= 'Month')+
    ylab('')+
  theme_minimal()+
    theme(axis.text.x = element_text(vjust = 0.5, hjust=1))+
    facet_wrap(~platform)+
    theme(text = element_text(size = 17),
    axis.text.x = element_text(angle = 90, hjust = 1))
```

```
P_Cla
```

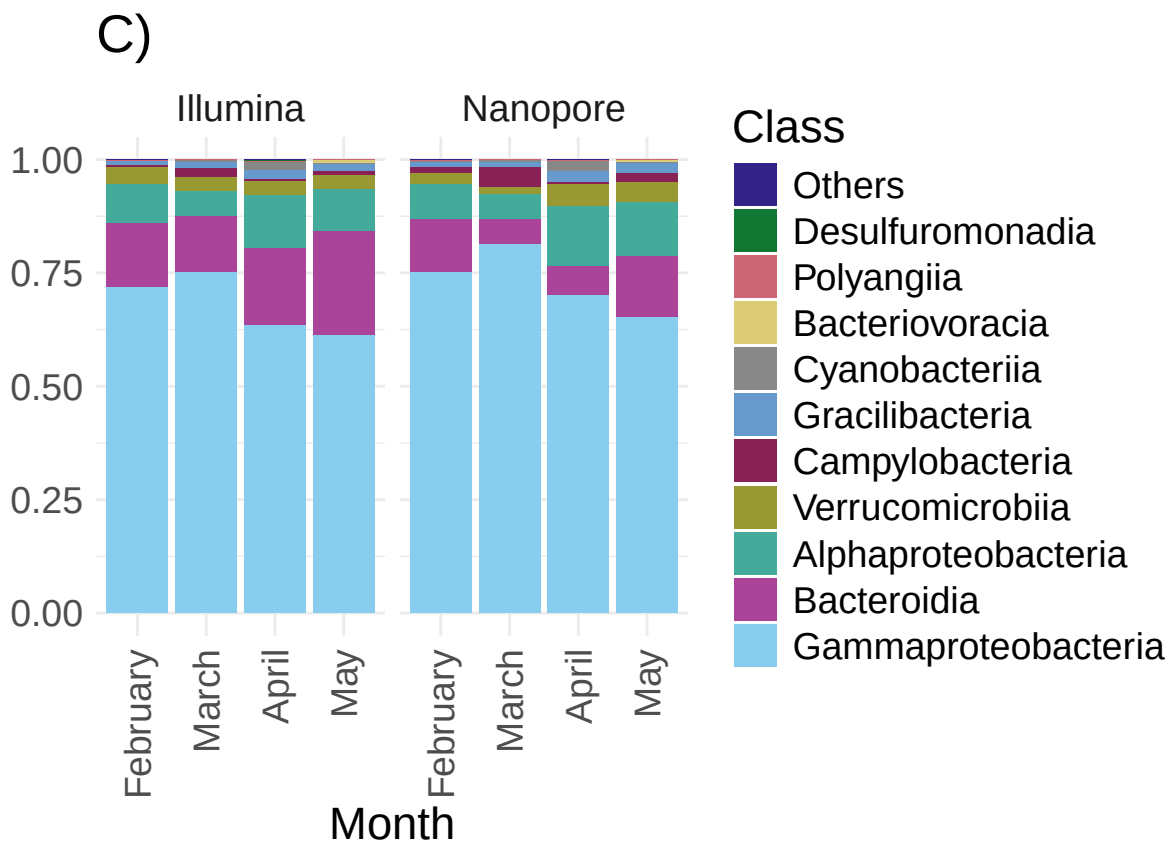

Figure 5: Microbiome composition: class level

Range of abundance values Illumina

```
tapply(Tax_Cla1$Abundance_r,Tax_Cla1$Class,mean)
```

```
##      Acidimicrobiia Alphaproteobacteria     Bacteriovoracia         Bacteroidia
##            0.008630             8.725955            0.227150           16.659815
##     Bdellovibrionia     Campylobacteria      Cyanobacteriia    Desulfuromonadia
```

```
##            0.042805            1.059165            0.564635            0.093860
## Gammaproteobacteria     Gracilibacteria          Polyangiia    Verrucomicrobiia
##           67.624530            1.364240            0.103070            3.211395
##       NB1_j_Inc_Sed
##            0.003100
```

```
range(Tax_Cla1[Tax_Cla1$Class=='Gammaproteobacteria',]$Abundance_r)
```

```
## [1] 50.5129 82.9024
```

```
range(Tax_Cla1[Tax_Cla1$Class=='Bacteroidia',]$Abundance_r)
```

```
## [1]  6.3289 27.5635
```

```
range(Tax_Cla1[Tax_Cla1$Class=='Alphaproteobacteria',]$Abundance_r)
```

```
## [1]  3.2685 21.1373
```

Nanopore

```
tapply(Tax_Cla2$Abundance_r,Tax_Cla2$Class,mean)
```

```
##      Alphaproteobacteria             Anaerolineae                  Bacilli
##                9.733235                0.004950                0.046305
##          Bacteriovoracia              Bacteroidia          Bdellovibrionia
##                0.176425                9.264355                0.027835
##          Campylobacteria          Cyanobacteriia         Desulfuromonadia
##                2.221870                0.795460                0.057695
##      Gammaproteobacteria          Gracilibacteria            Lentisphaeria
##               72.607750                1.533175                0.007495
##                   OM190            Parcubacteria               Polyangiia
##                0.009835                0.003080                0.113700
##         Verrucomicrobiia Marinimicrobia_inc._sed.
##                3.108980                0.002540
```

```
range(Tax_Cla2[Tax_Cla2$Class=='Gammaproteobacteria',]$Abundance_r)
```

```
## [1] 50.5034 87.8188
```

```
range(Tax_Cla2[Tax_Cla2$Class=='Bacteroidia',]$Abundance_r)
```

```
## [1]  3.4468 19.8849
```

```
range(Tax_Cla2[Tax_Cla2$Class=='Alphaproteobacteria',]$Abundance_r)
```

```
## [1]  3.4895 18.1953
```

11 Microbiome composition: order level

Exploration of orders in most abundant groups

```
sort(unique(Tax_Data1[Tax_Data1$Phylum=="p__Pseudomonadota",]$Order))
```

```
##  [1] "o__Arenicellales"
##  [2] "o__Azospirillales"
##  [3] "o__Beggiatoales"
##  [4] "o__Cardiobacteriales"
##  [5] "o__Caulobacterales"
##  [6] "o__Chromatiales"
##  [7] "o__Ectothiorhodospirales"
##  [8] "o__Enterobacterales"
##  [9] "o__Francisellales"
## [10] "o__Gammaproteobacteria_Incertae_Sedis"
## [11] "o__Granulosicoccales"
## [12] "o__Hyphomicrobiales"
## [13] "o__Kiloniellales"
## [14] "o__Micavibrionales"
## [15] "o__Parvibaculales"
## [16] "o__Pseudomonadales"
## [17] "o__Rhodobacterales"
## [18] "o__Rickettsiales"
## [19] "o__Sphingomonadales"
```

```
sort(unique(Tax_Data1[Tax_Data1$Phylum=="p__Pseudomonadota",]$Class))
```

```
## [1] "c__Alphaproteobacteria" "c__Gammaproteobacteria"
```

```
sort(unique(Tax_Data1[Tax_Data1$Phylum=="p__Bacteroidota",]$Order))
```

```
## [1] "o__Bacteroidales"      "o__Chitinophagales"    "o__Flavobacteriales"
## [4] "o__Sphingobacteriales"
```

```
sort(unique(Tax_Data1[Tax_Data1$Phylum=="p__Bacteroidota",]$Class))
```

```
## [1] "c__Bacteroidia"
```

Adjusting the dataset to order level Illumina

```
Tax_Data1[Tax_Data1$Order=="o__Incertae_Sedis",]$Order<-
               c('c__Gracilibacteria_Inc_Sed','p__NB1_j_Inc_Sed')

Tax_Order1<- aggregate(Tax_Data1[,2:21],by=list(Order=Tax_Data1$Order), sum)
Tax_Order1$Order<-substr(Tax_Order1$Order,4,40)

Tax_Order1<-data.frame(t(Tax_Order1))
colnames(Tax_Order1)<-Tax_Order1[1,]
Tax_Order1<-Tax_Order1[2:21,]
Tax_Order1$month<-substr(rownames(Tax_Order1),3,10)
Tax_Order1$Sample_ID<-rownames(Tax_Order1)
Tax_Order1$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Ord1<- gather(Tax_Order1,Order,Abundance,
                  Gracilibacteria_Inc_Sed:NB1_j_Inc_Sed, factor_key=TRUE)
Tax_Ord1$Abundance_r<-round(as.numeric(Tax_Ord1$Abundance),4)
```

Nanopore

```
Tax_Data2[Tax_Data2$Order=="o__Incertae_Sedis",]$Order<-c('c__OM190_inc._sed.',
            'c__Gracilibacteria_inc._sed.','p__Marinimicrobia_inc._sed.')

Tax_Order2<- aggregate(Tax_Data2[,2:21],by=list(Order=Tax_Data2$Order), sum)
Tax_Order2$Order<-substr(Tax_Order2$Order,4,40)

Tax_Order2<-data.frame(t(Tax_Order2))
colnames(Tax_Order2)<-Tax_Order2[1,]
Tax_Order2<-Tax_Order2[2:21,]
Tax_Order2$month<-substr(rownames(Tax_Order2),3,10)
Tax_Order2$Sample_ID<-rownames(Tax_Order2)
Tax_Order2$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Ord2<- gather(Tax_Order2,Order,Abundance,
                  Gracilibacteria_inc._sed.:Marinimicrobia_inc._sed., factor_key=TRUE)
Tax_Ord2$Abundance_r<-round(as.numeric(Tax_Ord2$Abundance),4)
```

Filtering only 10 orders with the highest abundance

```
Part<-names(sort(tapply(as.numeric(Tax_Ord1$Abundance),Tax_Ord1$Order,mean)))[28:36]
Tax_Ord1$Order_N<-as.character(Tax_Ord1$Order)
Tax_Ord1[!(Tax_Ord1$Order_N %in% Part),]$Order_N<-'Others'
Tax_Ord1$platform<-'Illumina'

Part<-names(sort(tapply(as.numeric(Tax_Ord2$Abundance),Tax_Ord2$Order,mean)))[33:41]
Tax_Ord2$Order_N<-as.character(Tax_Ord2$Order)
Tax_Ord2[!(Tax_Ord2$Order_N %in% Part),]$Order_N<-'Others'
Tax_Ord2$platform<-'Nanopore'
```

Combining datasets

```
Tax_Ord<-rbind(Tax_Ord1,Tax_Ord2)

Tax_Ord_New<-aggregate(Tax_Ord$Abundance_r,list(month=Tax_Ord$month,
            Order=Tax_Ord$Order_N,platform=Tax_Ord$platform),mean)

Tax_Ord_New[Tax_Ord_New$Order=='Gracilibacteria_inc._sed.',]$Order<-
  'Gracilibacteria inc. sed.'

Tax_Ord_New$month<-factor(Tax_Ord_New$month,levels=c('February','March','April','May'))
Tax_Ord_New$Order<-factor(Tax_Ord_New$Order,
    levels = c('Others','Gracilibacteria inc. sed.','Hyphomicrobiales',
        'Chitinophagales','Campylobacterales','Beggiatoales',
        'Verrucomicrobiales','Enterobacterales','Ectothiorhodospirales',
        'Rhodobacterales','Flavobacteriales','Cardiobacteriales'))

sort(tapply(Tax_Ord_New$x,Tax_Ord_New$Order,mean))
```

```
##                    Others Gracilibacteria inc. sed.          Hyphomicrobiales
##                 0.2033918                 1.4747300                 1.6160450
##           Chitinophagales         Campylobacterales              Beggiatoales
```

```
##                1.8190450                2.2218700                2.5550400
##       Verrucomicrobiales          Enterobacterales     Ectothiorhodospirales
##                3.1601875                3.4527925                3.5953125
##          Rhodobacterales          Flavobacteriales         Cardiobacteriales
##                7.5388050               11.6850825               58.2786300
```

Figure of bacterial compositon at order level

```
P_Ord<-ggplot(Tax_Ord_New,aes(x = month, y = x, fill = Order)) +
  geom_bar(position = "fill", stat = "identity") +
  scale_fill_manual(values = c("#DDCC77","#CC6677","#117733","#332288",
        '#993B3B',"#888888","#88CCEE","#AA4499","#44AA99","#999933",
        "#882255","#6699CC")) +
  labs(title= 'B)', x= 'Month')+
    ylab('')+
  theme_minimal()+
    theme(axis.text.x = element_text(vjust = 0.5, hjust=1))+
    facet_wrap(~platform)+
    theme(text = element_text(size = 17),
    axis.text.x = element_text(angle = 90, hjust = 1))
```

Combined figure

```
P_Ord
```

Exploration of mean abundance for individual groups

```
tapply(Tax_Ord1$Abundance_r,Tax_Ord1$Order,mean)
```

```
##            Gracilibacteria_Inc_Sed                         Arenicellales
##                           1.299950                             1.427025
##                     Azospirillales                    Bacteriovoracales
##                           0.008035                             0.227150
##                      Bacteroidales                    Bdellovibrionales
##                           0.002105                             0.042805
##                      Beggiatoales                       Bradymonadales
##                           2.661800                             0.093860
##                  Campylobacterales                    Cardiobacteriales
##                           1.059165                            52.589930
##                    Caulobacterales                     Chitinophagales
##                           0.171285                             1.819045
##                      Chloroplast                        Chromatiales
##                           0.564635                             0.014205
##              Ectothiorhodospirales                     Enterobacterales
##                           5.512815                             3.860580
##                   Flavobacteriales                      Francisellales
##                          14.763100                             0.042455
## Gammaproteobacteria_Incertae_Sedis                    Granulosicoccales
##                           0.642520                             0.036685
##                     Haliangiales                      Hyphomicrobiales
##                           0.033485                             1.616045
##                   JGI_0000069-P22                         Kiloniellales
```

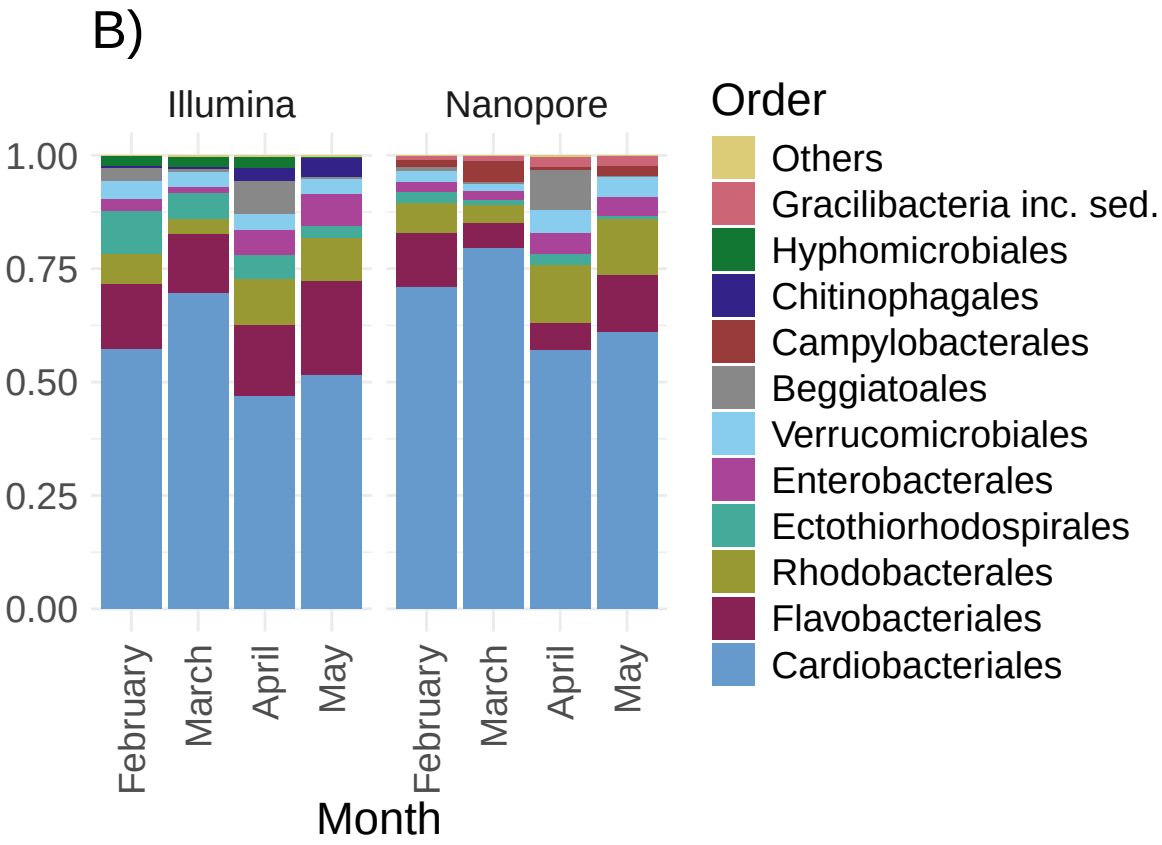

Figure 6: Microbiome composition: order level

```
##                         0.064300                         0.023525
##                    Micavibrionales                    Microtrichales
##                         0.003330                         0.008630
##                     Nannocystales                     Parvibaculales
##                         0.035025                         0.004755
##                     Polyangiales                    Pseudomonadales
##                         0.034570                         0.836540
##                    Rhodobacterales                    Rickettsiales
##                         6.755940                         0.025910
##                  Sphingobacteriales                 Sphingomonadales
##                         0.075580                         0.117125
##                 Verrucomicrobiales                   NB1_j_Inc_Sed
##                         3.211395                         0.003100
```

```
tapply(Tax_Ord2$Abundance_r,Tax_Ord2$Order,mean)
```

```
##          Gracilibacteria_inc._sed.                       OM190_inc._sed.
##                           1.474730                           0.009835
##                     Ardenticatenales                      Arenicellales
##                           0.004950                           0.790415
##                         Bacillales                   Bacteriovoracales
##                           0.002180                           0.176425
##                   Bdellovibrionales                       Beggiatoales
##                           0.027835                           2.448280
##                      Bradymonadales                     Burkholderiales
##                           0.057695                           0.001815
##                   Campylobacterales      Candidatus_Staskawiczbacteria
##                           2.221870                           0.003080
##                   Cardiobacteriales                     Caulobacterales
##                          63.967330                           0.124820
##                     Chitinophagales                        Chloroplast
##                           0.657305                           0.795460
##                        Chromatiales               Ectothiorhodospirales
##                           0.003040                           1.677810
##                    Enterobacterales                   Exiguobacterales
##                           3.045005                           0.008615
##                    Flavobacteriales                     Francisellales
##                           8.607065                           0.026305
## Gammaproteobacteria_Incertae_Sedis                  Granulosicoccales
##                           0.134335                           0.022475
##                       Haliangiales                   Hyphomicrobiales
##                           0.066180                           1.060510
##                    JGI_0000069-P22                      Kiloniellales
##                           0.058455                           0.012785
##                    Lactobacillales                    Lentisphaerales
##                           0.031955                           0.007495
##                    Micavibrionales                      Nannocystales
##                           0.004655                           0.022265
##                     Parvibaculales                       Polyangiales
##                           0.002430                           0.025250
##                    Pseudomonadales                    Rhodobacterales
##                           0.490935                           8.321670
##                      Rickettsiales                    Sphingomonadales
##                           0.096875                           0.109490
```

```
##                   Staphylococcales                 Verrucomicrobiales
##                           0.003550                           3.108980
##          Marinimicrobia_inc._sed.
##                           0.002540
```

12 Adjustment of dataset to genus level

Illumina

```
Tax_Data1$Genus_N<-paste0(Tax_Data1$Genus,'_',Tax_Data1$SampleID)

Tax_Data1[Tax_Data1$Genus_N=='g__Incertae_Sedis_sp1',]$Genus_N<-
  'g__Cardiobacteriaceae inc. sed._sp1'
Tax_Data1[Tax_Data1$Genus_N=="g__NS10_marine_group_sp6",]$Genus_N<-
  'g__Cryomorphaceae NS10 marine group_sp1'
Tax_Data1[Tax_Data1$Genus_N=="g__Incertae_Sedis_sp3",]$Genus_N<-
  'g__Ectothiorhodospiraceae inc. sed._sp1'
Tax_Data1[Tax_Data1$Genus_N=="g__Incertae_Sedis_sp11",]$Genus_N<-
  'g__Flavobacteriaceae inc. sed._sp1'
Tax_Data1[Tax_Data1$Genus_N=="g__Incertae_Sedis_sp8",]$Genus_N<-
  'g__Flavobacteriales NS9 inc. sed._sp1'
Tax_Data1[Tax_Data1$Genus_N=="g__Incertae_Sedis_sp10",]$Genus_N<-
  'g__Rhizobiaceae inc. sed._sp1'

Tax_Genus1<- aggregate(Tax_Data1[,2:21],by=list(Genus=Tax_Data1$Genus_N), sum)
Tax_Genus1$Genus<-substr(Tax_Genus1$Genus,4,40)
```

Nanopore

```
Tax_Data2$Genus_N<-paste0(Tax_Data2$Genus,'_',Tax_Data2$SampleID)

Tax_Data2[Tax_Data2$Genus_N=='g__Incertae_Sedis_sp1',]$Genus_N<-
  'g__Cardiobacteriaceae inc. sed._sp1'
Tax_Data2[Tax_Data2$Genus_N=="g__NS10_marine_group_sp6",]$Genus_N<-
  'g__Cryomorphaceae NS10 marine group_sp1'
Tax_Data2[Tax_Data2$Genus_N=="g__Incertae_Sedis_sp3",]$Genus_N<-
  'g__Ectothiorhodospiraceae inc. sed._sp1'
Tax_Data2[Tax_Data2$Genus_N=="g__Incertae_Sedis_sp11",]$Genus_N<-
  'g__Flavobacteriaceae inc. sed._sp1'
Tax_Data2[Tax_Data2$Genus_N=="g__Incertae_Sedis_sp8",]$Genus_N<-
  'g__Flavobacteriales NS9 inc. sed._sp1'
Tax_Data2[Tax_Data2$Genus_N=="g__Incertae_Sedis_sp10",]$Genus_N<-
  'g__Rhizobiaceae inc. sed._sp1'

Tax_Genus2<- aggregate(Tax_Data2[,2:21],by=list(Genus=Tax_Data2$Genus_N), sum)
Tax_Genus2$Genus<-substr(Tax_Genus2$Genus,4,40)
```

Tidying the data set Illumina

```
Tax_Genus1<-data.frame(t(Tax_Genus1))
colnames(Tax_Genus1)<-Tax_Genus1[1,]
```

```
Tax_Genus1<-Tax_Genus1[2:21,]
Tax_Genus1$month<-substr(rownames(Tax_Genus1),3,10)
Tax_Genus1$Sample_ID<-rownames(Tax_Genus1)
Tax_Genus1$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Gen1<- gather(Tax_Genus1,Genus,Abundance,Acinetobacter_sp196:Zobellia_sp171, factor_key=TRUE)
Tax_Gen1$Abundance_r<-round(as.numeric(Tax_Gen1$Abundance),4)
```

Nanopore

```
Tax_Genus2<-data.frame(t(Tax_Genus2))
colnames(Tax_Genus2)<-Tax_Genus2[1,]
Tax_Genus2<-Tax_Genus2[2:21,]
Tax_Genus2$month<-substr(rownames(Tax_Genus2),3,10)
Tax_Genus2$Sample_ID<-rownames(Tax_Genus2)
Tax_Genus2$Sample_ID<-paste0(rep(1:5,4),'_',rep(c('Feb','Mar','Apr','May'),each=5))

Tax_Gen2<- gather(Tax_Genus2,Genus,Abundance,Acinetobacter_sp196:Yoonia_sp23, factor_key=TRUE)
Tax_Gen2$Abundance_r<-round(as.numeric(Tax_Gen2$Abundance),2)
```

Filtering only 10 genus with the highest abundance

```
Part<-names(sort(tapply(as.numeric(Tax_Gen1$Abundance),Tax_Gen1$Genus,mean)))[124:132]
Tax_Gen1$Genus_N<-as.character(Tax_Gen1$Genus)
Tax_Gen1[!(Tax_Gen1$Genus_N %in% Part),]$Genus_N<-'Others'
Tax_Gen1$platform<-'Illumina'

Part<-names(sort(tapply(as.numeric(Tax_Gen2$Abundance),Tax_Gen2$Genus,mean)))[126:134]
Tax_Gen2$Genus_N<-as.character(Tax_Gen2$Genus)
Tax_Gen2[!(Tax_Gen2$Genus_N %in% Part),]$Genus_N<-'Others'
Tax_Gen2$platform<-'Nanopore'
```

Combining datasets

```
Tax_Gen<-rbind(Tax_Gen1,Tax_Gen2)

Tax_Gen_New<-aggregate(Tax_Gen$Abundance_r,list(month=Tax_Gen$month,
            Genus=Tax_Gen$Genus_N,platform=Tax_Gen$platform),mean)
```

Tidying the dataset

```
Tax_Gen_New1<-aggregate(Tax_Gen$Abundance_r,list(month=Tax_Gen$month,Sample=Tax_Gen$Sample_ID,
            Genus=Tax_Gen$Genus_N,platform=Tax_Gen$platform),mean)

New_Illum<-Tax_Gen_New1[Tax_Gen_New1$platform=='Illumina',]
New_Nan<-Tax_Gen_New1[Tax_Gen_New1$platform=='Nanopore',]
```

13 Venn diagrams: platforms

```
Tax_Phylum1<- aggregate(Tax_Data1[,2:21],by=list(Phylum=Tax_Data1$Phylum), sum)
Tax_Genus1<- aggregate(Tax_Data1[,2:21],by=list(Genus=Tax_Data1$Genus_N), sum)
Ill_Phyl<-substr(Tax_Phylum1$Phylum,4,30)
Ill_Gen<-substr(Tax_Genus1$Genus,4,30)

Tax_Phylum2<- aggregate(Tax_Data2[,2:21],by=list(Phylum=Tax_Data2$Phylum), sum)
Tax_Genus2<- aggregate(Tax_Data2[,2:21],by=list(Genus=Tax_Data2$Genus_N), sum)
Nan_Phyl<-substr(Tax_Phylum2$Phylum,4,20)
Nan_Gen<-substr(Tax_Genus2$Genus,4,30)
```

Calculate abundance

```
Tax_Gen_Il<-aggregate(as.numeric(Tax_Gen1$Abundance),list(Genus=Tax_Gen1$Genus),mean)
colnames(Tax_Gen_Il)[2]<-'Illumina'
Tax_Gen_Na<-aggregate(as.numeric(Tax_Gen2$Abundance),list(Genus=Tax_Gen2$Genus),mean)
colnames(Tax_Gen_Na)[2]<-'Nanopore'
```

Merging of datasets

```
Tax_Gen_O<-merge(Tax_Gen_Il,Tax_Gen_Na,by='Genus',all=TRUE)
Tax_Gen_O[is.na(Tax_Gen_O)]<-0
Tax_Gen_O$Overall_Abun<-rowMeans(Tax_Gen_O[,2:3])
```

Calculation of abundance for shared genera

```
x <- list(Illumina = Ill_Gen, Nanopore = Nan_Gen)
ItemsList <- gplots::venn(x, show.plot = FALSE)

Abun_list<-attributes(ItemsList)$intersections$`Illumina:Nanopore`
sum(Tax_Gen_O[Tax_Gen_O$Genus %in% Abun_list,]$Overall_Abun)
```

```
## [1] 29.8648
```

```
Tax_Gen_O$Genus_N<-Tax_Gen_O$Genus
Tax_Gen_O$Genus<-sub("_s[^_]+$", "", as.vector(Tax_Gen_O$Genus))
Tax_Gen_O$SampleID<-paste0('s',sub(".*_s", "", as.vector(Tax_Gen_O$Genus_N)))
```

Figure of venn diaggram for platforms

```
Venn_plat<-ggvenn(x,fill_color = c("#D55E00","#0072B2"),stroke_size = 0.5,
                  fill_alpha = 0.8,text_size = 5.5,
    show_percentage=FALSE,set_name_size = 0)+
    ggplot2::annotate(geom="text",x=0.06, y=-0.2,label="98.6 %",size = 5)+
    ggplot2::theme(text = element_text(size = 8),
        legend.position = "none",
        plot.title = element_blank())
```

```
Venn_plat <- Venn_plat + theme(plot.tag = element_text(size = 16, face = "plain"))

Venn_plat  +
  plot_annotation(tag_levels = list(c("A)")))
```

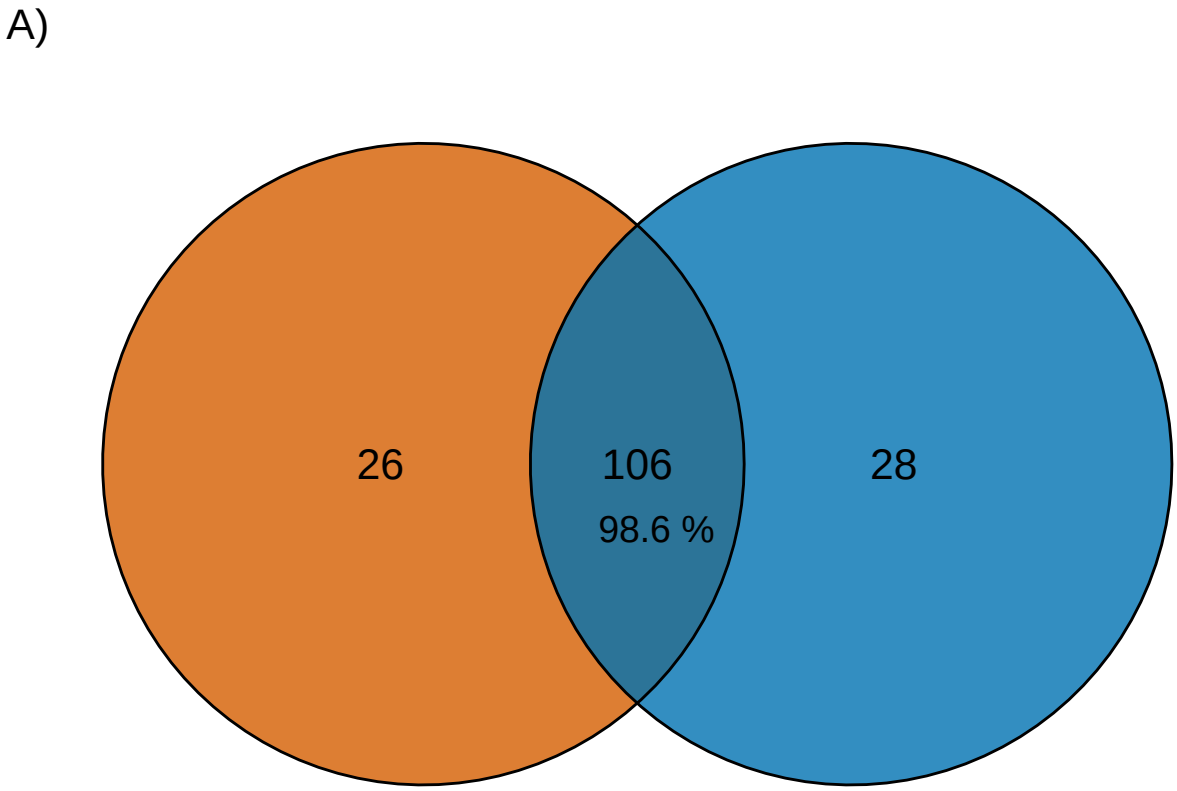

Figure 7: Venn diagram: platforms

14 Venn diagram: Comparison of months

```
Ill_Feb<-Tax_Genus1[,c(1:6)][apply(Tax_Genus1[,c(2:6)],1,sum)>0,]
Ill_Mar<-Tax_Genus1[,c(1,7:11)][apply(Tax_Genus1[,c(7:11)],1,sum)>0,]
Ill_Apr<-Tax_Genus1[,c(1,12:16)][apply(Tax_Genus1[,c(12:16)],1,sum)>0,]
Ill_May<-Tax_Genus1[,c(1,17:21)][apply(Tax_Genus1[,c(17:21)],1,sum)>0,]

Nan_Feb<-Tax_Genus2[,c(1:6)][apply(Tax_Genus2[,c(2:6)],1,sum)>0,]
Nan_Mar<-Tax_Genus2[,c(1,7:11)][apply(Tax_Genus2[,c(7:11)],1,sum)>0,]
Nan_Apr<-Tax_Genus2[,c(1,12:16)][apply(Tax_Genus2[,c(12:16)],1,sum)>0,]
Nan_May<-Tax_Genus2[,c(1,17:21)][apply(Tax_Genus2[,c(17:21)],1,sum)>0,]
```

Calculation of abundance

```
Tax_Gen_I<-aggregate(as.numeric(Tax_Gen1$Abundance),list(Genus=Tax_Gen1$Genus),mean)
colnames(Tax_Gen_I)[2]<-'Illumina'
Tax_Gen_N<-aggregate(as.numeric(Tax_Gen2$Abundance),list(Genus=Tax_Gen2$Genus),mean)
colnames(Tax_Gen_N)[2]<-'Nanopore'
```

Illumina

```
Illum <- list(February=Ill_Feb$Genus,March=Ill_Mar$Genus,April=Ill_Apr$Genus,
        May=Ill_May$Genus)

ItemsList <- gplots::venn(Illum, show.plot = FALSE)
Abun_list<-attributes(ItemsList)$intersections$`February:March:April:May`
Abun_list<-substr(Abun_list,4,30)
```

Nanopore

```
Nanop <- list(February=Nan_Feb$Genus,March=Nan_Mar$Genus,April=Nan_Apr$Genus,
        May=Nan_May$Genus)

ItemsList <- gplots::venn(Nanop, show.plot = FALSE)
Abun_list<-attributes(ItemsList)$intersections$`February:March:April:May`
Abun_list<-substr(Abun_list,4,30)
```

Figures of venn diagram for each platform separately

```
P1<-ggvenn(Illum,
  fill_color = c("#CC6677","#DDCC77","#117733","#332288"),
  stroke_size = 0.5,set_name_size = 2.8,fill_alpha = 0.5,
    show_percentage=FALSE)+
    ggplot2::ggtitle('A)  Illumina') +
    ggplot2::annotate(geom="text",x=0.06, y=-0.85,label="99.3 %")

P2<-ggvenn(Nanop,
  fill_color = c("#CC6677","#DDCC77","#117733","#332288"),
  stroke_size = 0.5,set_name_size = 2.8,fill_alpha = 0.5,
    show_percentage=FALSE)+
    ggplot2::ggtitle('B)  Nanopore') +
    ggplot2::annotate(geom="text",x=0.06, y=-0.85,label="99.5 %")
```

Both figures

```
(P1 | P2 )
```

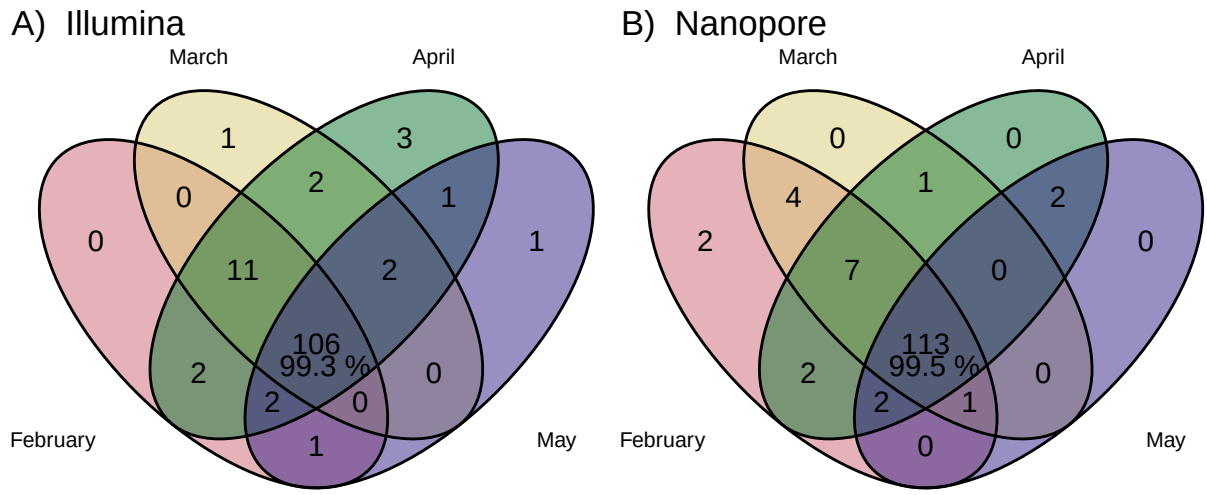

Figure 8: Venn diagram: months

15 LDA and LefSe analysis

Adjustment of data for analysis

```
colnames(TAX1)[c(1)]<-c('Kingdom')
Ill_phylo_filt_1 <- phyloseq(otu_table(Ill_atu_filt), TAX1, SAM1)

colnames(TAX2)[c(1)]<-c('Kingdom')
Nan_phylo_filt_1 <- phyloseq(otu_table(Nan_atu_filt), TAX2, SAM2)
```

LefSe analysis

```
result_Illum <- run_lefse(Ill_phylo_filt_1,group='time', taxa_rank='Genus')
result_Nano <- run_lefse(Nan_phylo_filt_1,group='time', taxa_rank='Genus')
```

Compile results into a single dataset

```
LDA_Illu<-data.frame(Group=marker_table(result_Illum)$feature,
                     Month=marker_table(result_Illum)$enrich_group,
    LDA=marker_table(result_Illum)$ef_lda,Platform='Illumina')

LDA_Nan<-data.frame(Group=marker_table(result_Nano)$feature,
                    Month=marker_table(result_Nano)$enrich_group,
    LDA=marker_table(result_Nano)$ef_lda,Platform='Nanopore')

LDA<-rbind(LDA_Nan,LDA_Illu)
```

Filtering of groups with LDA scores higher than 4

```
LDA_filt <- LDA[ave(LDA$LDA >= 4, LDA$Group, FUN = any), ]

LDA_filt$Group<-substr(LDA_filt$Group,4,50)
LDA_filt$Group<-factor(LDA_filt$Group,levels=c(sort(unique(LDA_filt$Group))))
```

Adjustment of dataset for plotting

```
levels(LDA_filt$Group)[c(2:7,10:12)]<-c("Cardiobacteriaceae inc. sed.",
    "Cyanobacteriia chloroplast inc. sed.","Cryomorphaceae inc. sed.",
    "Ectothiorhodospiraceae inc. sed.","Flavobacteriaceae inc. sed.",
    "Gracilibacteria inc. sed.","Cryomorphaceae NS10 marine group",
    "Flavobacteriales NS9 inc. sed.","Rhizobiaceae inc. sed."
    )

LDA_filt_compl <- LDA_filt %>%
  complete(Group, Platform, fill = list(LDA = 0))
```

Figure of LefSe analysis

```
LefSe<-ggplot(LDA_filt_compl, aes(LDA, Group, fill = Platform)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.85), width = 0.85) +
  scale_fill_manual(values = c("#D55E00", "#0072B2")) +
  labs(y = '', x = 'LDA score [log10]') +
  theme_minimal() +
  theme(text = element_text(size = 15)) +
  geom_vline(xintercept = 4, linetype = "dashed") +
  scale_y_discrete(labels = c(
    expression(italic("Arcobacter")),
    expression(italic("Cardiobacteriaceae inc. sed.")),
      expression("Cyanobacteriia chloroplast " * italic("inc. sed.")),
    expression(italic("Cryomorphaceae inc. sed.")),
    expression(italic("Ectothiorhodospiraceae inc. sed.")),
    expression(italic("Flavobacteriaceae inc. sed.")),
    expression(italic("Gracilibacteria inc. sed.")),
    expression(italic("Halocynthiibacter")),
    expression(italic("Leucothrix")),
      expression(italic("Cryomorphaceae") * " NS10 marine group"),
      expression("Flavobacteriales NS9 " * italic("inc. sed.")),
    expression(italic("Rhizobiaceae inc. sed.")),
    expression(italic("Rubritalea")),
```

```
    expression(italic("Ruegeria")),
    expression(italic("Salinimicrobium")),
    expression(italic("Tenacibaculum")),
    expression(italic("Vibrio"))))
```

The final figure together with venn diagram of platforms

```
LefSe    <- LefSe + theme(plot.tag = element_text(size = 16, face = "plain"))

(LefSe) +
  plot_annotation(tag_levels = list(c("B)")))
```

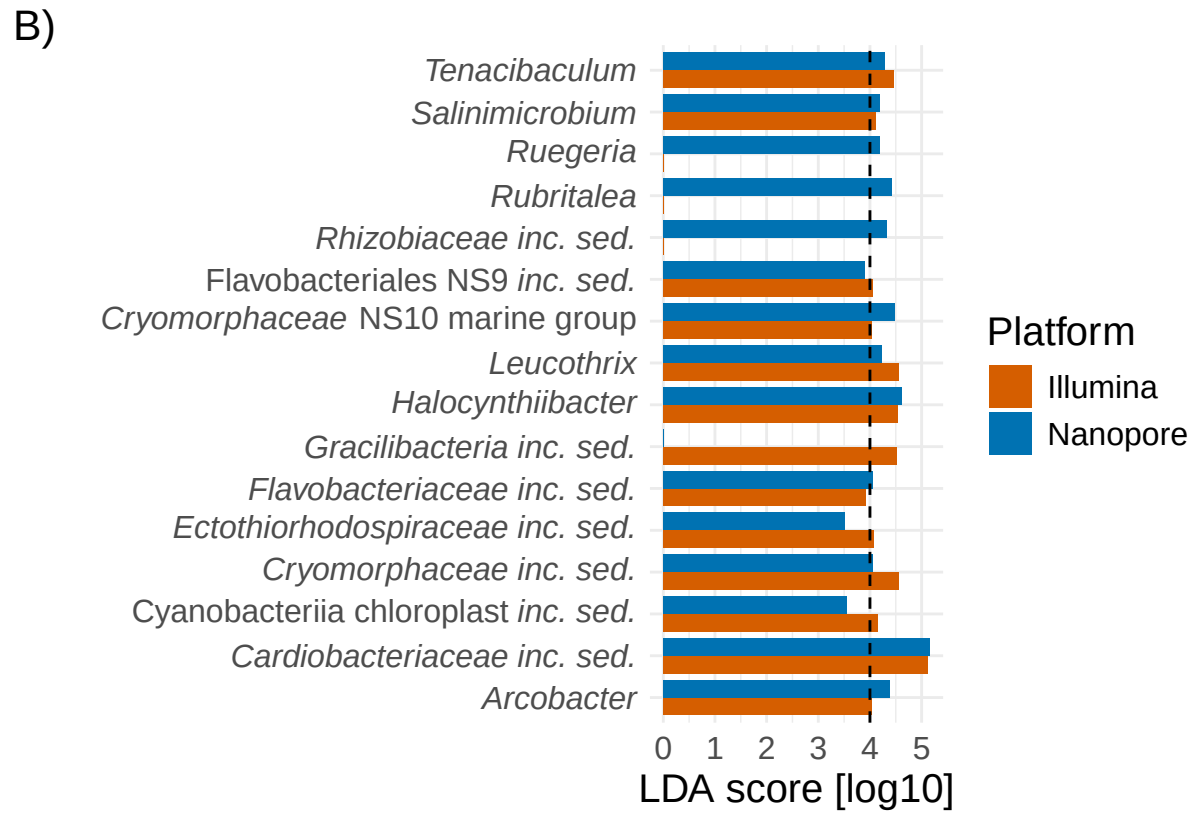

Figure 9: LefSe analysis

16 Identification of pathogen-associated genera

```
Detected<-sort(substr(unique(observation$Genus),4,25))
```

List of pathogen-associated genera

```
Pathogens<-c('Acinetobacter','Aerococcus','Aeromonas','Aliivibrio','Aquaspirillum',
        'Arcobacter','Carnobacterium','Chlamydia','Chryseobacterium',
        'Citrobacter','Clostridium','Delftia','Edwardsiella','Enterobacter',
        'Erysipelothrix','Escherichia','Flavobacterium','Flectobacillus',
        'Francisella','Hafnia','Hahella','Halomonas','Janthinobacterium',
        'Klebsiella','Lactococcus','Moritella','Mycobacterium','Mycoplasma',
        'Nocardia','Pantoea','Pasteurella','Photobacterium','Piscirickettsia',
        'Plesiomonas','Providencia','Pseudoalteromonas','Pseudomonas',
        'Renibacterium','Salmonella','Serratia','Shewanella','Stenotrophomonas',
        'Streptococcus','Tenacibaculum','Vagococcus','Vibrio','Weissella','Yersinia')
```

Matching detected genera to the list of fish pathogens

```
pattern<- c(Pathogens[1])
myStrings <- observation$Genus
Det_Pathogens<-data.frame()

Det_Pathogens<-observation[observation$Genus %in% c(myStrings[grepl(pattern, myStrings)]),]

for (i in 2:48){
    pattern<- c(Pathogens[i])
    myStrings <- observation$Genus
    if(length(myStrings[grepl(pattern, myStrings)])>0){
    Part<-observation[observation$Genus %in% c(myStrings[grepl(pattern, myStrings)]),]
    Det_Pathogens<-rbind(Det_Pathogens,Part)
    }}

Det_Pathogens$Genus_N<-substr(Det_Pathogens$Genus,4,20)
```

Matching the list of detected pathogens to abundance data at genus level

```
Tax_Gen_New1<-aggregate(Tax_Gen$Abundance_r,list(Month=Tax_Gen$month,
            Genus=Tax_Gen$Genus,platform=Tax_Gen$platform),mean)
colnames(Tax_Gen_New1)[4]<-'mean'
Tax_Gen_New2<-aggregate(Tax_Gen$Abundance_r,list(Month=Tax_Gen$month,
            Genus=Tax_Gen$Genus,platform=Tax_Gen$platform),function(x) {sd(x)/sqrt(length(x))})
colnames(Tax_Gen_New2)[4]<-'se'
Tax_Gen_New<-merge(Tax_Gen_New1,Tax_Gen_New2,by=c('Month','Genus','platform'),all=TRUE)

Tax_Gen_New$Genus_N<-sub("_s[^_]+$", "", as.vector(Tax_Gen_New$Genus))
Tax_Gen_Pat<- Tax_Gen_New[Tax_Gen_New$Genus_N %in% c(Det_Pathogens$Genus_N),]

sort(tapply(Tax_Gen_Pat$mean,Tax_Gen_Pat$Genus_N,sum))
```

```
##     Streptococcus           Pantoea          Serratia      Enterobacter
##           0.01000           0.01200           0.01400           0.01400
##       Citrobacter        Halomonas       Pseudomonas     Acinetobacter
##           0.01600           0.02400           0.18526           0.20584
##        Shewanella       Aliivibrio       Francisella Pseudoalteromonas
##           0.21684           0.22970           0.27582           0.50954
##     Tenacibaculum           Vibrio        Arcobacter
##           9.77840          12.81272          13.12066
```

```
Tax_Gen_Pat1<-Tax_Gen_Pat[Tax_Gen_Pat$Genus_N %in% c('Tenacibaculum','Vibrio','Arcobacter'),]

Tax_Gen_Pat1$Genus_N<-factor(Tax_Gen_Pat1$Genus_N,
    levels = c('Tenacibaculum','Vibrio','Arcobacter'))
Tax_Gen_Pat1$Month<-factor(Tax_Gen_Pat1$Month,
    levels = c('February','March','April','May'))
```

Testing of differences in abundance between months

```
Abun<-aggregate(Tax_Gen$Abundance_r,list(Month=Tax_Gen$month,sample=Tax_Gen$Sample_ID,
            Genus=Tax_Gen$Genus,platform=Tax_Gen$platform),mean)
Abun$Genus_N<-sub("_s[^_]+$", "", as.vector(Abun$Genus))
Abun_Pat<- Abun[Abun$Genus_N %in% c(Det_Pathogens$Genus_N),]
Abun_Pat1<-Abun_Pat[Abun_Pat$Genus_N %in% c('Tenacibaculum','Vibrio','Arcobacter'),]

Abun_Pat1$Devide<-paste(Abun_Pat1$Genus_N,Abun_Pat1$platform)

for(i in 1:length(unique(Abun_Pat1$Devide))){
    A<-Abun_Pat1[Abun_Pat1$Devide %in% unique(Abun_Pat1$Devide)[i],]
    Krus<-kruskal.test(x ~ Month, data = A)
    if(Krus$p.value<0.05){
        print(unique(Abun_Pat1$Devide)[i])
        print(pairwise.wilcox.test(A$x,A$Month,p.adjust.method = "BH"))
            }else{
        print('NO')}}
```

```
## [1] "Arcobacter Illumina"
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
##
## data:  A$x and A$Month
##
##          April February March
## February 1.000 -        -
## March    0.095 0.167    -
## May      0.190 0.208    0.208
##
## P value adjustment method: BH
## [1] "Tenacibaculum Illumina"
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
##
## data:  A$x and A$Month
##
##          April February March
## February 0.421 -        -
## March    0.143 0.048    -
## May      0.143 0.181    0.048
##
## P value adjustment method: BH
## [1] "Vibrio Illumina"
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
```

```
##
## data:  A$x and A$Month
##
##          April February March
## February 0.226 -        -
## March    0.095 0.267    -
## May      0.421 0.190    0.048
##
## P value adjustment method: BH
## [1] "Arcobacter Nanopore"
##
##  Pairwise comparisons using Wilcoxon rank sum exact test
##
## data:  A$x and A$Month
##
##          April February March
## February 0.548 -        -
## March    0.024 0.143    -
## May      0.024 0.181    0.143
##
## P value adjustment method: BH
## [1] "Tenacibaculum Nanopore"
##
##  Pairwise comparisons using Wilcoxon rank sum test with continuity correction
##
## data:  A$x and A$Month
##
##          April February March
## February 0.346 -        -
## March    0.226 0.346    -
## May      0.016 0.016    0.016
##
## P value adjustment method: BH
## [1] "Vibrio Nanopore"
##
##  Pairwise comparisons using Wilcoxon rank sum test with continuity correction
##
## data:  A$x and A$Month
##
##          April February March
## February 0.402 -        -
## March    0.117 0.141    -
## May      0.402 0.095    0.072
##
## P value adjustment method: BH
```

```
Ill_Arc<-c('a','a','a','a')
Ill_Ten<-c('a','b','ab','a')
Ill_Vib<-c('ab','a','ab','b')
Nan_Arc<-c('ab','a','b','a')
Nan_Ten<-c('a','a','a','b')
Nan_Vib<-c('a','a','a','a')
```

Preparation of data for plotting

```
summary_data <- Tax_Gen_Pat1 %>%
  group_by(platform,Genus_N,Month) %>%
  summarise(mean = mean(mean, na.rm = TRUE),
            se = se,.groups = 'drop')

summary_data <- summary_data %>%
    mutate(
        Month = factor(Month, levels = c("February", "March", "April", "May")),
        Genus_N = factor(Genus_N, levels = sort(unique(Genus_N))),
        platform = factor(platform, levels = c("Illumina", "Nanopore")) ) %>%
            arrange(platform, Genus_N, Month)

label_data <- summary_data %>%
  mutate(label = c(Ill_Ten,Ill_Vib,Ill_Arc,Nan_Ten,Nan_Vib,Nan_Arc),
         y = mean + se + 0.15)
```

```
Abun_Pat1$Genus_N<-factor(Abun_Pat1$Genus_N,levels=levels(Tax_Gen_Pat1$Genus_N))
Abun_Pat1$Month<-factor(Abun_Pat1$Month,levels=c('February','March','April','May'))

Tax_Gen_Pat1$month<-factor(Tax_Gen_Pat1$Month,levels=c("February", "March", "April", "May"))
label_data<-as.data.frame(label_data)
label_data$Month<-factor(label_data$Month,levels=c("February", "March", "April", "May"))

Illumina<-Tax_Gen_Pat1[Tax_Gen_Pat1$platform=='Illumina',]
label_d<-label_data[label_data$platform=='Illumina',]
Illumina$Genus_N<-factor(Illumina$Genus_N,levels=c('Arcobacter','Tenacibaculum','Vibrio'))
label_data$Genus_N<-factor(Illumina$Genus_N,levels=c('Arcobacter','Tenacibaculum','Vibrio'))
```

Figure of pathogenic-genera abundance

```
ggplot(Illumina,aes(x =Genus_N, y = mean, color = Month)) +
    geom_point(position=position_dodge(width = .5),size=4)  +
    labs(x= 'Genus',y= 'Proportion of the total abundance')+
    theme_minimal()+
    geom_errorbar(aes(ymin=mean-se, ymax=mean+se), width=.4,
                  position=position_dodge(.5)) +
    scale_color_manual(values = c("#CC6677","#DDCC77","#88CCEE","#117733")) +
#   geom_point(data = Abun_Pat1,
#              mapping = aes(x = Genus_N, y = x, color = Month),
#       position=position_dodge(width = .5),size=1)+
    geom_text(data = label_d,
              aes(x = Genus_N, y = mean+se+0.5, label = label,color=Month,group=Month),
              position = position_dodge(width = 0.5),
              inherit.aes = FALSE,
              size = 5)+
    theme(text = element_text(size = 17),
        axis.text.x = element_text(face = "italic"))
```

17 List of detected pathogen-associated genera

Using the original, unfiltered dataset, the pathogen-associated genera were identified

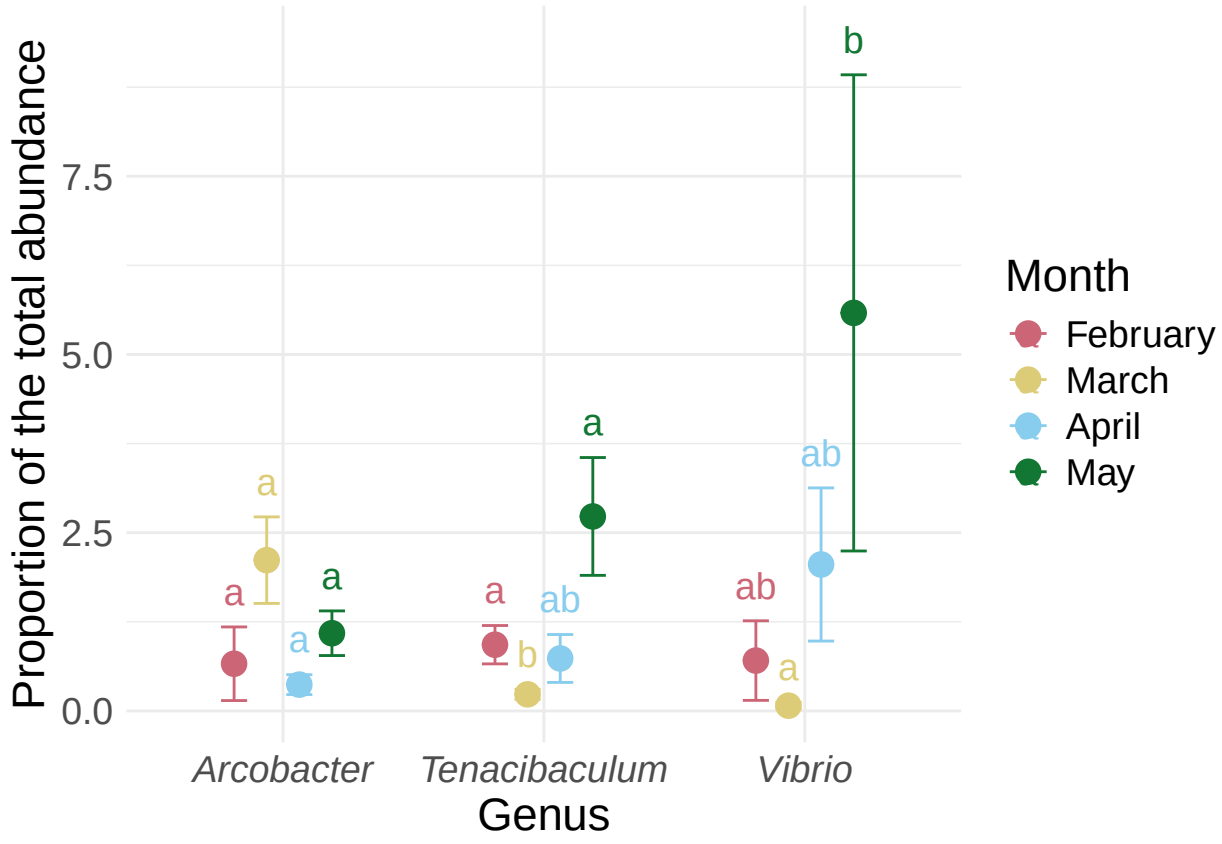

Figure 10: Abundance of pathogenic genera

```
Sequen$Sample_ID<-rownames(Sequen)
Count_Path<-Sequen[Sequen$Sample_ID %in% c(Det_Pathogens$SampleID),]

Count_Path<-data.frame(t(Count_Path[,1:40]))
Count_Path$Month<-c(rep('Feb',5),rep('Mar',5),rep('Apr',5),rep('May',5))
Count_Path$Platform<-c(rep('Illumina',20),rep('Nanopore',20))
Count_Path<- aggregate(Count_Path[,1:32],by=list(Platform=Count_Path$Platform), sum)

Tax<-observation[observation$SampleID %in% colnames(Count_Path),]
colnames(Count_Path)[2:33]<-Tax$Genus

Count_Path_F<- gather(Count_Path,Genus,Count,g__Arcobacter:g__Plesiomonas, factor_key=TRUE)
Count_Path_F$Genus<-substr(Count_Path_F$Genus,4,20)
```

Filtering for genera with at least 10 reads in all samples

```
genus_platform <- with(Count_Path_F, tapply(Count, list(Genus, Platform), sum))
keep_genera <- rownames(genus_platform)[apply(genus_platform >= 10, 1, any)]
Count_Path_F3<-Count_Path_F[Count_Path_F$Genus %in% keep_genera, ]

Count_Path_F4<- spread(Count_Path_F3,Platform,Count)
Count_Path_F4
```

```
##                Genus Illumina Nanopore
## 1      Acinetobacter      754      291
## 2          Aeromonas        1       29
## 3         Aliivibrio      753       75
## 4         Arcobacter    17838    85966
## 5     Carnobacterium        0       32
## 6   Chryseobacterium       59       95
## 7        Citrobacter        0      160
## 8       Enterobacter        0      145
## 9     Flavobacterium       51       60
## 10       Francisella      707     1163
## 11         Halomonas       38      222
## 12        Klebsiella        0       27
## 13         Moritella       14       14
## 14           Pantoea       10      118
## 15    Photobacterium       29       89
## 16 Pseudoalteromonas     1606     1239
## 17       Pseudomonas      477      849
## 18          Serratia       14      174
## 19        Shewanella      515      899
## 20     Streptococcus        7      131
## 21     Tenacibaculum    17152    50583
## 22        Vagococcus        0       11
## 23            Vibrio    29992    40976
## 24          Yersinia        0       25
```

Number of detected genera

```r
nrow(Count_Path_F4)
```

```
## [1] 24
```

```r
nrow(Count_Path_F4[Count_Path_F4$Illumina>10,])
```

```
## [1] 15
```

```r
nrow(Count_Path_F4[Count_Path_F4$Nanopore>10,])
```

```
## [1] 24
```

Calculation of overall abundance for pathogen-associated genera

```r
Selec<-Tax_Data[Tax_Data$SampleID %in% Tax$SampleID,]
Selec_Ill<-Selec[,c(2:21)]
rownames(Selec_Ill)<-Selec[,c(47)]
Selec_Nan<-Selec[,c(22:41)]
rownames(Selec_Nan)<-Selec[,c(47)]

apply(Selec_Ill,1,mean)
```

```
## g__Pseudoalteromonas        g__Shewanella       g__Pseudomonas
##         0.0953781261         0.0317194305         0.0268116334
##         g__Halomonas          g__Serratia       g__Citrobacter
##         0.0021288701         0.0008060716         0.0000000000
##      g__Enterobacter     g__Acinetobacter     g__Streptococcus
##         0.0000000000         0.0439498559         0.0004541397
##     g__Tenacibaculum           g__Pantoea            g__Vibrio
##         1.1565005355         0.0006220995         2.1044095166
##       g__Francisella        g__Aliivibrio        g__Arcobacter
##         0.0424416578         0.0574088603         1.0590923918
```

```r
apply(Selec_Nan,1,mean)
```

```
## g__Pseudoalteromonas        g__Shewanella       g__Pseudomonas
##          0.031319935          0.022574648          0.020239272
##         g__Halomonas          g__Serratia       g__Citrobacter
##          0.006116891          0.004114234          0.003911157
##      g__Enterobacter     g__Acinetobacter     g__Streptococcus
##          0.003545888          0.007892926          0.003245913
##     g__Tenacibaculum           g__Pantoea            g__Vibrio
##          1.288451901          0.002747797          1.098207789
##       g__Francisella        g__Aliivibrio        g__Arcobacter
##          0.026300369          0.001758507          2.221776247
```

18 Species level identification of pathogens

Loading data: Microbiome pathogens

Table 1: Abundance of putative pathogens

	Species	Ab_Total	Ab_Feb	Ab_Mar	Ab_Apr	Ab_May
1	Acinetobacter johnsonii	0.0006533	NA	NA	0.0026131	NA
5	Aeromonas sobria	0.0003459	0.0013838	NA	NA	NA
6	Aliivibrio logei	0.0004137	NA	NA	NA	0.0016549
7	Aliivibrio wodanis	0.0005598	NA	NA	NA	0.0022393
15	Moritella marina	0.0003134	NA	0.0012535	NA	NA
24	Pseudomonas fluorescens	0.0053724	NA	0.0214895	NA	NA
25	Pseudomonas koreensis	0.0047448	NA	0.0189790	NA	NA
35	Tenacibaculum dicentrarchi	1.1136566	0.4381292	0.3672788	0.7267405	2.9224779
37	Tenacibaculum maritimum	0.0549587	0.0095763	0.0291027	0.0238874	0.1572685
38	Tenacibaculum ovolyticum	0.1026456	0.1153723	0.0315391	0.0153900	0.2482810
39	Tenacibaculum soleae	0.0045132	0.0087597	0.0071975	NA	0.0020955
41	Vibrio alginolyticus	0.0100154	0.0209182	NA	0.0191432	NA
42	Vibrio anguillarum	0.0007944	NA	0.0031777	NA	NA
46	Vibrio splendidus	0.3163324	0.1338102	0.0080011	0.3384830	0.7850352
47	Vibrio tapetis	0.0096269	NA	0.0271711	0.0064678	0.0048689

```
Pathog<-read.table("Sealice_Microbiome_Pathogens.csv",
    header = TRUE,sep = ';',fill = TRUE,dec = ",",na.strings = "NA")
```

Calculation of abundance for individual pathogens and months

```
Pathog<-Pathog[Pathog$Species %in% c('Acinetobacter johnsonii','Aeromonas sobria',
        'Aliivibrio logei','Aliivibrio wodanis','Moritella marina',
        'Pseudomonas fluorescens','Pseudomonas koreensis',
        'Tenacibaculum dicentrarchi','Tenacibaculum maritimum',
        'Tenacibaculum ovolyticum','Tenacibaculum soleae',
        'Vibrio alginolyticus','Vibrio anguillarum','Vibrio splendidus',
        'Vibrio tapetis'),]


Patogen<-data.frame(Species=Pathog$Species,Ab_Total=apply
                    (Pathog[,c(3:22)],1,mean),
    Ab_Feb=apply(Pathog[,c(3:7)],1,mean),Ab_Mar=apply
    (Pathog[,c(8:12)],1,mean),
    Ab_Apr=apply(Pathog[,c(13:17)],1,mean),Ab_May=apply
    (Pathog[,c(18:22)],1,mean))

Patogen[Patogen$Ab_Feb==0,]$Ab_Feb<-NA
Patogen[Patogen$Ab_Mar==0,]$Ab_Mar<-NA
Patogen[Patogen$Ab_Apr==0,]$Ab_Apr<-NA
Patogen[Patogen$Ab_May==0,]$Ab_May<-NA
```

```
knitr::kable(Patogen)
```