Analysis set2 updated

Marko Suokas

Libraries

```
library(mia);packageVersion("mia")
```

```
[1] '1.12.0'
```

```
library(vegan);packageVersion("vegan")
```

```
[1] '2.6.6.1'
```

```
library(scater);packageVersion("scater")
```

```
[1] '1.32.1'
```

```
library(tidyverse);packageVersion("tidyverse")
```

```
[1] '2.0.0'
```

```
library(kableExtra);packageVersion("kableExtra")
```

```
[1] '1.4.0'
```

```
library(patchwork);packageVersion("patchwork")
```

```
[1] '1.2.0'
```

```
library(ggthemes);packageVersion("ggthemes")
```

```
[1] '5.1.0'
```

Import

```
#Rds files are easy way to save ja load data objects
tse_dada <- readRDS("set2/tse_dada.rds")
tse_vs97 <- readRDS("set2/tse_vs97.rds")
tse_vs99 <- readRDS("set2/tse_vs99.rds")
tse_emu <- readRDS("set2/tse_emu.rds")
```

Agglomeration of objects to genus level.

```
#Agglomerate
tse_dada<- agglomerateByRank(tse_dada, rank = "Genus", onRankOnly = T,
                             na.rm = F)
tse_vs97 <- agglomerateByRank(tse_vs97, rank = "Genus", onRankOnly = T,
                             na.rm = F)
tse_vs99 <- agglomerateByRank(tse_vs99, rank = "Genus", onRankOnly = T,
                             na.rm = F)
tse_emu <- agglomerateByRank(tse_emu, rank = "Genus", onRankOnly = T,
                             na.rm = F)
#Check number of variants
nrow(tse_dada)
```

```
[1] 39
```

```
nrow(tse_vs97)
```

```
[1] 51
```

```
nrow(tse_vs99)
```

```
[1] 49
```

```
nrow(tse_emu)
```

```
[1] 39
```

Calculate relative abundance for assay data

```
tse_dada <- transformAssay(tse_dada, assay.type = "counts",
                                   method = "relabundance")
tse_vs97 <- transformAssay(tse_vs97, assay.type = "counts",
                                   method = "relabundance")
tse_vs99 <- transformAssay(tse_vs99, assay.type = "counts",
                                   method = "relabundance")
tse_emu <- transformAssay(tse_emu, assay.type = "counts",
                                   method = "relabundance")
```

Getting top10 features

```
top10_dada <- getTopFeatures(tse_dada, top = 10, method = "mean",
                             assay.type = "relabundance")
#Fetch assay table and filter it
dada_table <- data.frame(assays(tse_dada)$relabundance)
dada_table <- dada_table %>% rownames_to_column(var = "Genus") %>%
  filter(Genus %in% top10_dada)
kable(dada_table, digits=2, caption = "Dada2 denoiser") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 11) %>%
  row_spec(0, background = "teal", color = "white")
```

Table 1: Dada2 denoiser

Genus	barcode07	barcode08	barcode09	barcode10	barcode11	barcode12
Acidocella	0.09	0.00	0.00	0.53	0.00	0.00
Acidithiobacillus	0.05	0.04	0.05	0.04	0.03	0.05
Microbacter	0.00	0.11	0.00	0.21	0.12	0.00
Fonticella	0.09	0.00	0.00	0.00	0.16	0.00
Thiomonas	0.00	0.00	0.00	0.00	0.17	0.10
Lachnoclostridium	0.00	0.00	0.23	0.00	0.00	0.14
Pelosinus	0.16	0.18	0.28	0.00	0.00	0.12
Desulfosporosinus	0.32	0.47	0.15	0.21	0.37	0.44
Mobilitalea	0.06	0.00	0.03	0.00	0.10	0.00
Herbinix	0.17	0.02	0.01	0.00	0.00	0.00

```
top10_vs97 <- getTopFeatures(tse_vs97, top = 10, method = "mean",
                             assay.type = "relabundance")
#Fetch assay table and filter it
vs97_table <- data.frame(assays(tse_vs97)$relabundance)
vs97_table <- vs97_table %>% rownames_to_column(var = "Genus") %>%
  filter(Genus %in% top10_vs97)
kable(vs97_table, digits=2, caption = "Vsearch 97") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 11) %>%
  row_spec(0, background = "teal", color = "white")
```

Table 2: Vsearch 97

Genus	barcode07	barcode08	barcode09	barcode10	barcode11	barcode12
Acidocella	0.03	0.01	0.00	0.52	0.00	0.01
Microbacter	0.00	0.13	0.00	0.25	0.14	0.00
Acidithiobacillus	0.06	0.05	0.06	0.04	0.03	0.05
Fonticella	0.09	0.00	0.00	0.00	0.15	0.00
Thiomonas	0.01	0.00	0.02	0.00	0.17	0.10
Desulfosporosinus	0.27	0.35	0.24	0.16	0.30	0.48
Pelosinus	0.21	0.24	0.26	0.02	0.01	0.08
Lachnoclostridium	0.00	0.00	0.24	0.00	0.00	0.15
Herbinix	0.23	0.01	0.00	0.00	0.00	0.00
Mobilitalea	0.01	0.01	0.00	0.00	0.11	0.00

```
top10_vs99 <- getTopFeatures(tse_vs99, top = 10, method = "mean",
                             assay.type = "relabundance")
#Fetch assay table and filter it
vs99_table <- data.frame(assays(tse_vs99)$relabundance)
vs99_table <- vs99_table %>% rownames_to_column(var = "Genus") %>%
  filter(Genus %in% top10_vs99)
kable(vs99_table, digits=2, caption = "Vsearch 99") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 11) %>%
  row_spec(0, background = "teal", color = "white")
```

Table 3: Vsearch 99

Genus	barcode07	barcode08	barcode09	barcode10	barcode11	barcode12
Acidocella	0.09	0.00	0.00	0.53	0.00	0.01
Microbacter	0.02	0.11	0.01	0.17	0.13	0.01
Acidithiobacillus	0.06	0.05	0.06	0.04	0.03	0.06
Fonticella	0.09	0.00	0.00	0.00	0.16	0.00
Thiomonas	0.00	0.00	0.00	0.01	0.19	0.06
Desulfosporosinus	0.27	0.37	0.17	0.18	0.30	0.41
Pelosinus	0.14	0.24	0.28	0.04	0.01	0.14
Lachnoclostridium	0.01	0.01	0.23	0.01	0.01	0.15
Herbinix	0.20	0.02	0.01	0.00	0.00	0.01
Mobilitalea	0.03	0.01	0.03	0.00	0.10	0.03

```
top10_emu <- getTopFeatures(tse_emu, top = 10, method = "mean",
                             assay.type = "relabundance")
#Fetch assay table and filter it
emu_table <- data.frame(assays(tse_dada)$relabundance)
emu_table <- emu_table %>% rownames_to_column(var = "Genus") %>%
  filter(Genus %in% top10_emu)
kable(emu_table, digits=2, caption = "Emu") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 11) %>%
  row_spec(0, background = "teal", color = "white")
```

Table 4: Emu

Genus	barcode07	barcode08	barcode09	barcode10	barcode11	barcode12
Acidocella	0.09	0.00	0.00	0.53	0.00	0.00
Acidithiobacillus	0.05	0.04	0.05	0.04	0.03	0.05
Microbacter	0.00	0.11	0.00	0.21	0.12	0.00
Fonticella	0.09	0.00	0.00	0.00	0.16	0.00
Thiomonas	0.00	0.00	0.00	0.00	0.17	0.10
Lachnoclostridium	0.00	0.00	0.23	0.00	0.00	0.14
Pelosinus	0.16	0.18	0.28	0.00	0.00	0.12
Desulfosporosinus	0.32	0.47	0.15	0.21	0.37	0.44
Herbinix	0.17	0.02	0.01	0.00	0.00	0.00
Anaerovorax	0.02	0.04	0.00	0.00	0.03	0.01

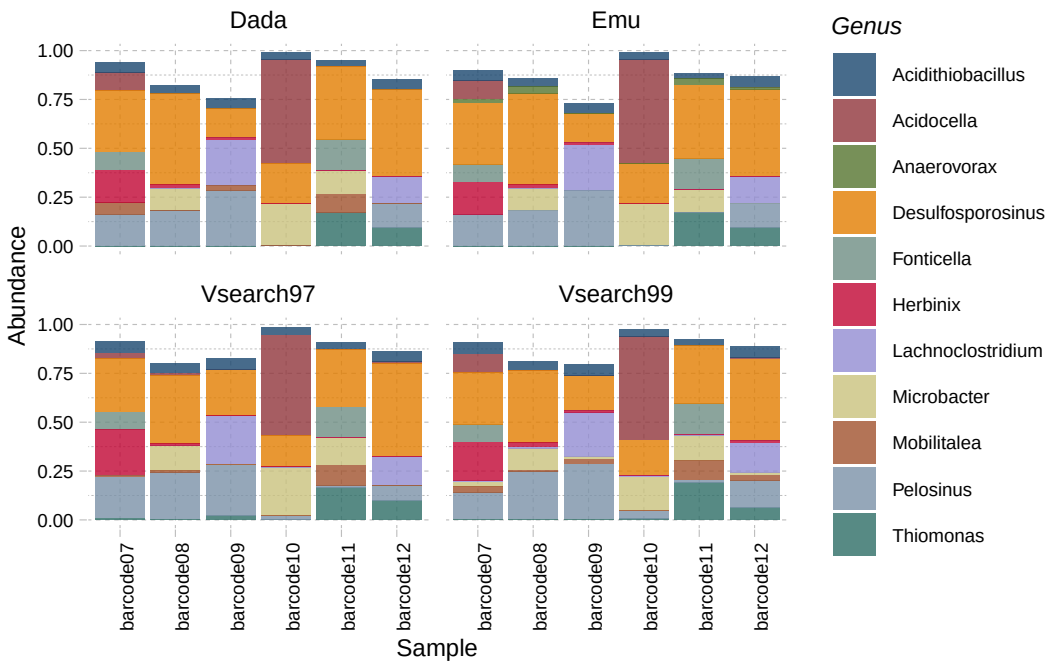
For barplot it is necessary to create long table

```
#transform data to ggplot
dada_long <- dada_table %>% pivot_longer(cols = starts_with("barcode"),
                                         names_to = "Sample",
                                         values_to = "Abundance") %>%
  mutate(Method = "Dada")
#transform data to ggplot
vs97_long <- vs97_table %>% pivot_longer(cols = starts_with("barcode"),
                                         names_to = "Sample",
                                         values_to = "Abundance") %>%
  mutate(Method = "Vsearch97")
#transform data to ggplot
vs99_long <- vs99_table %>% pivot_longer(cols = starts_with("barcode"),
                                         names_to = "Sample",
                                         values_to = "Abundance") %>%
  mutate(Method = "Vsearch99")
#transform data to ggplot
emu_long <- emu_table %>% pivot_longer(cols = starts_with("barcode"),
                                         names_to = "Sample",
                                         values_to = "Abundance") %>%
  mutate(Method = "Emu")
#combine
long_table <- bind_rows(dada_long, vs97_long, vs99_long, emu_long)
```

Plot object

```
#Create stacked barplot
ab_plot <- ggplot(long_table, aes(x = Sample, y = Abundance, fill = Genus)) +
  geom_bar(stat = "identity", alpha=0.8) + facet_wrap(~Method) +
  theme_pander(base_size = 9) + scale_fill_stata() +
  theme(axis.text.x = element_text(angle = 90))
```

```
ab_plot
```



Alpha diversity measured by Shannon index

```
#dada
tse_dada <- readRDS("set2/tse_dada.rds")
tse_dada<- transformAssay(tse_dada, assay.type = "counts", method = "relabundance",
                          name = "relabundance")
tse_dada <- estimateDiversity(tse_dada, assay.type="relabundance", index="shannon")
#vsearch97
tse_vs97 <- readRDS("set2/tse_vs97.rds")
tse_vs97 <- transformAssay(tse_vs97, assay.type = "counts", method = "relabundance",
                          name = "relabundance")
tse_vs97 <- estimateDiversity(tse_vs97, assay.type="counts", index="shannon")
#vsearch99
tse_vs99 <- readRDS("set2/tse_vs99.rds")
tse_vs99 <- transformAssay(tse_vs99, assay.type = "counts", method = "relabundance",
                          name = "relabundance")
tse_vs99 <- estimateDiversity(tse_vs99, assay.type="counts",index="shannon")
#emu
tse_emu <- readRDS("set2/tse_emu.rds")
tse_emu <- transformAssay(tse_emu, assay.type = "counts", method = "relabundance",
                          name = "relabundance")
tse_emu <- estimateDiversity(tse_emu, assay.type="counts",index="shannon")

#combine results
alpha <- data.frame(dada = colData(tse_dada)$shannon,
                    vsearch97 = colData(tse_vs97)$shannon,
                    vsearch99 = colData(tse_vs99)$shannon,
                    emu = colData(tse_emu)$shannon)
```

```
#table
kable(alpha, digits=2, caption = "Shannon index") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 11) %>%
  row_spec(0, background = "teal", color = "white")
```

Table 5: Shannon index

	dada	vsearch97	vsearch99	emu
barcode07	3.91	3.19	4.12	2.13
barcode08	4.31	3.31	4.04	1.98
barcode09	4.05	2.45	3.72	2.20
barcode10	2.58	1.87	2.17	1.24
barcode11	3.76	2.70	3.00	1.83
barcode12	4.05	2.47	4.57	2.12

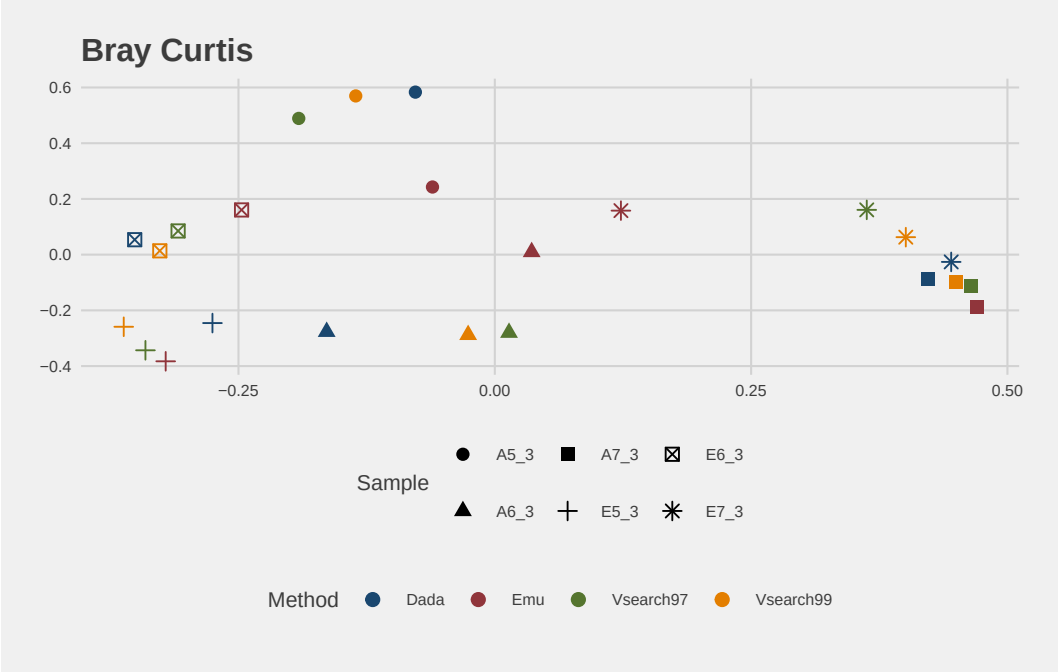
Beta diversity measured by Bray-Curtis dissimilarity

```
#create bray-curtis distance matrix
tse_dada <- runMDS(tse_dada, FUN = vegan::vegdist, method = "bray",
                   name="PCoA_BC", exprs_values = "relabundance")
dada_bray <- plotReducedDim(tse_dada, "PCoA_BC")
#create dataframe for plot
bray_dada_df <- data.frame(pcoa1 = dada_bray$data[,1],
                           pcoa2 = dada_bray$data[,2],
                           Sample = colData(tse_dada)$Name)
tse_vs99 <- runMDS(tse_vs99, FUN = vegan::vegdist, method = "bray",
                   name="PCoA_BC", exprs_values = "relabundance")
vsearch99_bray <- plotReducedDim(tse_vs99, "PCoA_BC")
#create dataframe for plot
bray_vsearch99_df <- data.frame(pcoa1 = vsearch99_bray$data[,1],
                           pcoa2 = vsearch99_bray$data[,2],
                           Sample = colData(tse_vs99)$Name)
tse_vs97 <- runMDS(tse_vs97, FUN = vegan::vegdist, method = "bray",
                   name="PCoA_BC", exprs_values = "relabundance")
vsearch97_bray <- plotReducedDim(tse_vs97, "PCoA_BC")
#create dataframe for plot
```

```
bray_vsearch97_df <- data.frame(pcoa1 = vsearch97_bray$data[,1],
                          pcoa2 = vsearch97_bray$data[,2],
                          Sample = colData(tse_vs97)$Name)
tse_emu <- runMDS(tse_emu, FUN = vegan::vegdist, method = "bray",
             name="PCoA_BC", exprs_values = "relabundance")
emu_bray <- plotReducedDim(tse_emu, "PCoA_BC")
#create dataframe for plot
bray_emu_df <- data.frame(pcoa1 = emu_bray$data[,1],
                          pcoa2 = emu_bray$data[,2],
                          Sample = colData(tse_emu)$Name)
#add method for each dataframe and combine
bray_dada_df$Method <- "Dada"
bray_vsearch99_df$Method <- "Vsearch99"
bray_vsearch97_df$Method <- "Vsearch97"
bray_emu_df$Method <- "Emu"
combined_pcoa <- bind_rows(bray_dada_df, bray_vsearch99_df, bray_vsearch97_df, bray_emu_df)
#create plot object
plot_pcoa <- ggplot(data = combined_pcoa, aes(x=pcoa1, y=pcoa2,
                                              color = Method, shape = Sample)) +
  labs(title = "Bray Curtis",
       x = "pcoa1", y = "pcoa2") + geom_point(size=2) +
  theme_fivethirtyeight(base_size=8) + scale_color_stata()
```

Plot pcoa

```
plot_pcoa
```



Observations

The overall results are generally consistent across methods, though it is important to note that the diversity in the analyzed samples is lower than what is typically found in most biological samples.

Clustering methods generate a higher number of variants compared to denoising, but the most abundant microbes remain consistent across methods. Interestingly, denoising yields a slightly higher Shannon index in 4 out of 6 samples. In the Bray-Curtis dissimilarity analysis, only one out of six samples shows a notable difference when denoising is applied.

The Emu profiler produced the smallest number of taxa, with only 92 identified. This lower diversity is also reflected in the Shannon index. In the Bray-Curtis plot, Emu displays noticeable differences compared to other methods, although the extent of this variation depends on the sample.

Given the current data, it may be prudent to use vsearch clustering for long amplicons until more comprehensive data is available on the performance of other methods in more complex datasets. A 99% identity level is recommended, along with filtering the least abundant variants to manage data size effectively. In this data set, there are 2800 "OTUs" with more than 10 counts globally.