Wk 6 - coding exercise answers

Christine Hawkes

2/16/2022

Coding Exercises

Please submit as a knitted html markdown to GitHub due on 2/23

##1. run PCoA on clr transformed otus in phyloseq

- Use `microbiome::transform` for clr transform on ps object
  - This function adds a pseudocount if there are zeros in data
  - Resulting transform will differ from `compositions::clr`
- Run ordination via `phyloseq::ordinate` with option `PCoA`
  - Use euclidean distance on clr == Aitchison's distance
  - Note: alternatively calculate Aitchison's distance via `robCompositions::aDist function`
- Use `phyloseq::plot_scree` to evaluate variance explained by each axis
- Plot ordination results
- Access sample scores and eigenvalues for export

```
# transform
ps_clr <- microbiome::transform(ps_gp_bact, transform="clr")
# ordination
ord_clr1 <- phyloseq::ordinate(ps_clr, "PCoA", "euclidean")
# scree plot - first 2 axes explain ~45% of total variation
phyloseq::plot_scree(ord_clr1)
# ord plot
phyloseq::plot_ordination(ps_clr, ord_clr1,
            type="sample", color="SampleType",
            label="SampleType",
            title="Phyloseq::PCoA clr plot")
# export sample scores
write.csv(ord_clr1$vectors, "Wk6_PCoA_xy1.csv")
# export eigenvalues
write.csv(ord_clr1$values, "Wk6_PCoA_eig.csv")
```

##2. run PCoA with clr transformed otus + environmental data using an alternative method

- Select a method for PCoA outside of phyloseq
  - Examples: `vegan::wcmdscale`,`vegan::rda`, `FactoMineR::PCA`, `ade4::dudi.pca`, `stats::prcomp`, `stats::princomp`, `ecodist::pco`, `ape::pcoa`

- Examine eigenvalues

- Plot results

- Describe how the clr-based ordination results here and in #1 above differ from the vst results and what this means for analysis

- Access sample scores and eigenvalues for export

```
# retrieve otu matrix
otu_clr <- otu_table(ps_clr)
otu_clr <- as.data.frame(t(otu_clr))

# calculate dissimilarities
dist_clr <- vegan::vegdist(otu_clr, "euclidean")

# PCoA in vegan using wcmdscale
ord_clr2 <- vegan::wcmdscale(dist_clr, k=2, eig=TRUE)
ord_clr2
summary(vegan::eigenvals(ord_clr2))

# envfit
ord_clr2_env <- vegan::envfit(ord_clr2, sam.new, permutations = 99, strata = NULL, choices=c(1,2))
ord_clr2_env

# many plot options
plot(ord_clr2)
ggordiplots::gg_ordiplot(ord=ord_clr2, groups=sam.old$SampleType,
                         kind = "sd", conf = 0.95, pt.size=4)
ggordiplots::gg_envfit(ord=ord_clr2, env=sam.new, perm=99,
                       pt.size=4, alpha= 0.1)
ggordiplots::gg_ordisurf(ord=ord_clr2, env.var=sam.new$pH,
                         binwidth=0.5, pt.size=1, var.label="pH")
ggordiplots::gg_ordisurf(ord=ord_clr2, env.var=sam.new$salinity,
                         binwidth=10, pt.size=1, var.label="salinity")

# accessing results for export
# sample scores
ord_clr2$points
# eigenvalues
ord_clr2$eig
# correlations
ord_clr2_env$vectors
```

##3. Examine beta-diversity in a phylogenetic context using DPCoA

- This will take ~10 min to run
  - if longer, consider further reducing GP dataset to top 100-200 taxa
- Analyze the Global Patterns bacteria data with DPCoA in `phyloseq::DPCoA`
  - https://rdrr.io/bioc/phyloseq/man/DPCoA.html
- Access the resulting list object using `$` to get the Axis 1 and 2 coordinates
- Use `data.frame` to combine the following into one file for plotting:
  - sample names from `SAM$X.SampleID`
  - Axis1 and Axis2 coordinates
  - sample types from `SAM$SampleType`
- Plot eigenvalues by axis with `phyloseq::plot_scree`
- Plot results by samples with `plot_ordination` using `color="SampleType"`
- Plot results by species with `plot_ordination` using `color="Phylum"`

```
# run DPCoA
ord_dpcoa <- phyloseq::DPCoA(ps_gp_bact)
# alternative
# ord_dpcoa <- ordinate(ps_gp_bact, "DPCoA")

# extract data from resulting ord5 list for plotting
str(ord_dpcoa)
Axis1 <- ord_dpcoa$li[["Axis1"]]
Axis2 <- ord_dpcoa$li[["Axis2"]]
ord_dpcoa_data <- data.frame(SAM$X.SampleID, Axis1, Axis2, SAM$SampleType)
# can write the above to csv

# eig plot
phyloseq::plot_scree(ord_dpcoa)
# samples plot
phyloseq::plot_ordination(ps_gp_bact, ord_dpcoa,
                          type="samples", color="SampleType") +
                          ggplot2::geom_point(size=4)
# species plot
phyloseq::plot_ordination(ps_gp_bact, ord_dpcoa,
                          type="species", color="Phylum", title="DPCoA") +
                          ggplot2::geom_point(size=4)
# note: I used separate plots to get around the limit of 6 shapes
```