MB590 Microbiome Analysis

Christine V. Hawkes

3/30/2022

Contents

Analyzing Factorial Designs with Permutation Procedures

```
References:

Collyer & Adams (2018) RRPP: An r package for fitting linear models to high-dimensional data using resi

Bolker et al. (2009) Generalized linear mixed models: a practical guide for ecology and evolution. TREE

Data:

Erlandson et al. (2018) Soil abiotic variables are more important than Salicaceae phylogeny or habitat s
DRYAD entry: https://datadryad.org/stash/dataset/doi:10.5061/dryad.5f24ks4
```

Libraries and Data

Install and load R libraries

```
#install.packages("RRPP")

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

```
## [1] '1.3.1'
```

```
library(phyloseq); packageVersion("phyloseq")
```

```
## [1] '1.38.0'
```

```
library(DESeq2); packageVersion("DESeq2")
```

```
## [1] '1.34.0'
```

```
library(RRPP); packageVersion("RRPP")
```

```
## [1] '1.1.2'
```

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

```
## [1] '2.5.7'
```

```
library(ggplot2); packageVersion("ggplot2")
```

```
## [1] '3.3.5'
```

```
library(ggordiplots); packageVersion("ggordiplots")
```

```
## [1] '0.4.0'
```

Load and prepare data

- Data from Erlandson et al. 2018 that we have used previously
- All files are on GitHub, add the raw url path to the read commands
- Or, if you saved the RData as suggested last week, you can open your own file

```
# load data
load("wk12_data.RData")
ps_vst
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 6758 taxa and 215 samples ]
## sample_data() Sample Data:       [ 215 samples by 41 sample variables ]
## tax_table()   Taxonomy Table:    [ 6758 taxa by 7 taxonomic ranks ]
```

```
# check that "Observed" is in your sample_data from last week's richness calcs
colnames(phyloseq::sample_data(ps_vst))
```

```
##  [1] "GardenID"           "Garden.Location"    "Number"
##  [4] "Treatment"          "June"               "July"
##  [7] "Aug"                "Mean"               "Nmin"
## [10] "NO3"                "NH4"                "pH"
## [13] "Spp"                "Ecology"            "Sample"
## [16] "Genotype"           "Caged.E..Not.Caged" "Plant_Height_m"
## [19] "Date_Sampled"       "extraction_date"    "Lat"
## [22] "Long"               "Plot"               "Dist1"
## [25] "Dist2"              "Dist3"              "order"
## [28] "TLP"                "WD"                 "SPI"
## [31] "LSV"                "RER"                "SLA"
## [34] "RGR"                "TLP.F"              "WD.F"
## [37] "SPI.F"              "SLA.F"              "Axis.1"
## [40] "Axis.2"             "Observed"
```

```
# useful functions to pull sample and otu files from the ps object in the correct formats
# phyloseq to dataframe
ps2df_sam <- function(physeq) {
  sd <- phyloseq::sample_data(physeq)
  return(as(sd,"data.frame"))
}

# phyloseq to matrix
ps2m_otu <- function(physeq) {
  OTU <- phyloseq::otu_table(physeq)
  if(phyloseq::taxa_are_rows(OTU)) {
    OTU <- t(OTU)
  }
  return(as(OTU, "matrix"))
}

# get data using above functions
SAM <- ps2df_sam(ps_vst)
OTU <- ps2m_otu(ps_vst)

# confirm that the two files have the same rownames
all(rownames(SAM)==rownames(OTU))
```

```
## [1] TRUE
```

Factorial analysis of alpha diversity

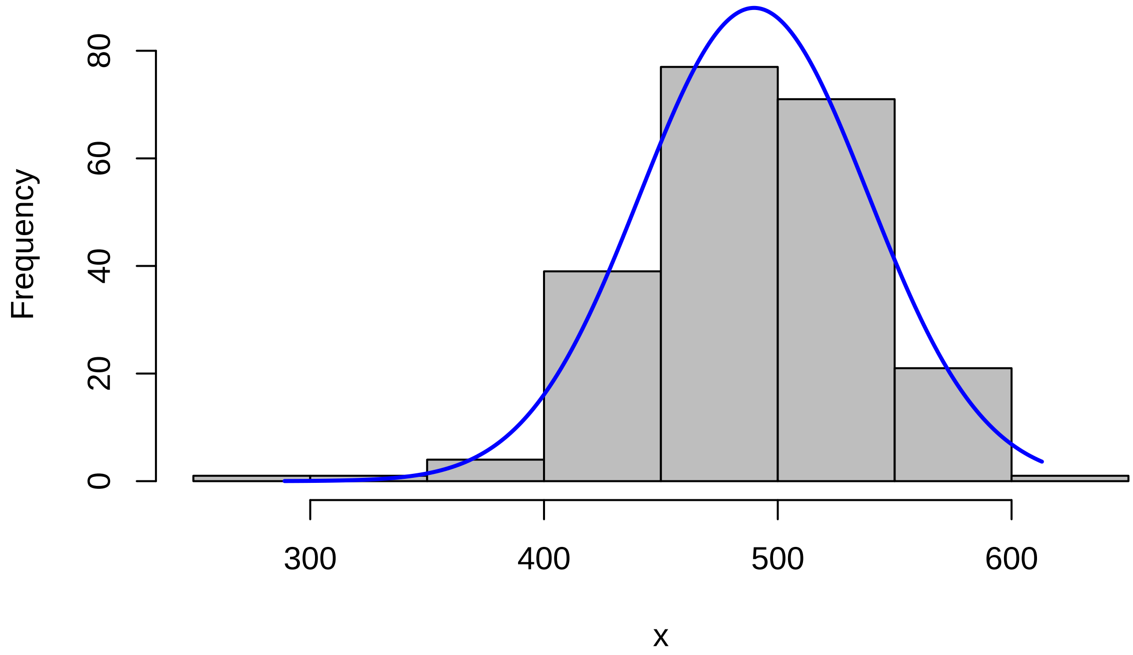
Check assumptions

```
# we'll use the non-parametric RRPP because some assumptions are violated

# test null hyp that sample comes from a normal distribution
# slightly off from normal
# note that sqrt transform from orig paper makes it worse!
shapiro.test((SAM$Observed))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  (SAM$Observed)
## W = 0.98329, p-value = 0.01204
```

```
rcompanion::plotNormalHistogram(SAM$Observed)
```



```
# test null hyp of no difference in variance across groups
# homogeneous variances except for Plot
bartlett.test(Observed ~Treatment, data=SAM)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Observed by Treatment
## Bartlett’s K-squared = 0.03395, df = 1, p-value = 0.8538
```

```
bartlett.test(Observed ~Spp, data=SAM)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Observed by Spp
## Bartlett’s K-squared = 14.871, df = 13, p-value = 0.3155
```

```
bartlett.test(Observed ~Plot, data=SAM)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Observed by Plot
## Bartlett’s K-squared = 26.409, df = 12, p-value = 0.009391
```

permANOVA - richness

RRPP - richness fixed effects model

```
# define dependent var
rich <- SAM$Observed

# fixed factor only - ignores random terms
# with this model, Treatment has a significant effect on richness
rich.rrpp <- RRPP::lm.rrpp(rich ~ Treatment,
                   data = SAM, SS.type="III",
                   print.progress = FALSE, iter=1000)
anova(rich.rrpp, effect.type = "F")
```

```
##
## Analysis of Variance, using Residual Randomization
## Permutation procedure: Randomization of null model residuals
## Number of permutations: 1001
## Estimation method: Ordinary Least Squares
## Sums of Squares and Cross-products: Type III
## Effect sizes (Z) based on F distributions
##
##            Df     SS      MS     Rsq      F      Z   Pr(>F)
## Treatment   1  30587 30587.4 0.06014 13.629 3.0473 0.000999 ***
## Residuals 213 478024  2244.2 0.93986
## Total     214 508611
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Call: RRPP::lm.rrpp(f1 = rich ~ Treatment, iter = 1000, SS.type = "III",
##     data = SAM, print.progress = FALSE)
```

RRPP - richness mixed model

```
# orig paper used Spp and Plot as a random effects
# plot here does not include interaction given replication issues
# rerun RRPP as mixed model with both
# sig spatial effect of Plot and only a trend for Treatment
rich.rrpp2 <- RRPP::lm.rrpp(rich ~ Treatment*Spp+Plot,
                    data = SAM, SS.type="III",
                    print.progress = FALSE, iter=1000)
```

```
##
## Warning: Because variables in the linear model are redundant,
## the linear model design has been truncated (via QR decomposition).
## Original X columns: 29
## Final X columns (rank): 28
## Check coefficients or degrees of freedom in ANOVA to see changes.
```

```
# if you don't specify the MS error terms, model will use Residuals
anova(rich.rrpp2, effect.type = "F")
```

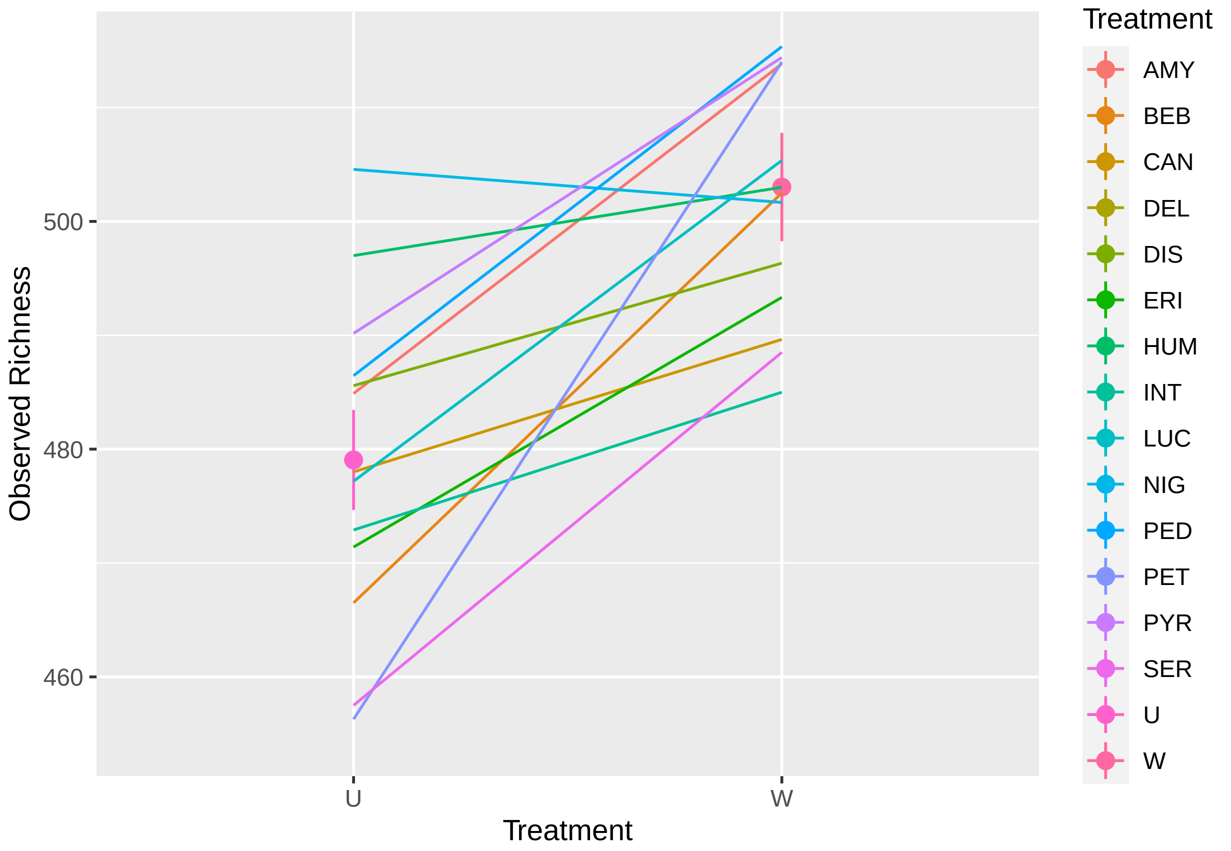
```
##
## Analysis of Variance, using Residual Randomization
## Permutation procedure: Randomization of null model residuals
## Number of permutations: 1001
## Estimation method: Ordinary Least Squares
## Sums of Squares and Cross-products: Type III
## Effect sizes (Z) based on F distributions
##
##                 Df     SS    MS     Rsq       F       Z   Pr(>F)
## Treatment        1   3444  3444 0.00677  1.6361  0.8313 0.218781
## Spp             13  21254  1635 0.04179  0.7766 -0.5073 0.691309
## Plot             1  54296 54296 0.10675 25.7904  3.6303 0.000999 ***
## Treatment:Spp   12  10446   871 0.02054  0.4135 -1.7091 0.958042
## Residuals      187 393687  2105 0.77404
## Total          214 508611
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Call: RRPP::lm.rrpp(f1 = rich ~ Treatment * Spp + Plot, iter = 1000,
##     SS.type = "III", data = SAM, print.progress = FALSE)
```

```
# to get correct F ratios, specify MS error terms
# check order from model output
anova(rich.rrpp2, effect.type = "F",
      error = c("Treatment:Spp", "Residuals", "Residuals", "Residuals"))
```

```
##
## Analysis of Variance, using Residual Randomization
## Permutation procedure: Randomization of null model residuals
## Number of permutations: 1001
## Estimation method: Ordinary Least Squares
## Sums of Squares and Cross-products: Type III
## Effect sizes (Z) based on F distributions
##
##                 Df     SS    MS     Rsq       F       Z   Pr(>F)
## Treatment        1   3444  3444 0.00677  3.9567  1.5709 0.051948 .
## Spp             13  21254  1635 0.04179  0.7766 -0.5073 0.691309
## Plot             1  54296 54296 0.10675 25.7904  3.6303 0.000999 ***
## Treatment:Spp   12  10446   871 0.02054  0.4135 -1.7091 0.958042
## Residuals      187 393687  2105 0.77404
## Total          214 508611
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Call: RRPP::lm.rrpp(f1 = rich ~ Treatment * Spp + Plot, iter = 1000,
##     SS.type = "III", data = SAM, print.progress = FALSE)
```

Plot richness data for interpretation

```
# use a reaction norm plot to view how Spp range between Trtmt gardens
ggplot2::ggplot(SAM, ggplot2::aes(x=Treatment, y=rich, color=Treatment)) +
  ggplot2::stat_summary(fun.data="mean_se", geom="pointrange") +
  ggplot2::stat_summary(ggplot2::aes(group = Spp, color=Spp), fun = "mean", geom = "path") +
  ggplot2::ylab("Observed Richness")
```



Factorial analysis of beta diversity

PermANOVA - beta diversity

```
# Note: can use mvnormtest::mshapiro.test for multivariate Shapiro-Wilks
# but only works for <5000 OTUs

# OLS
# For real data, typically use iter=1000
# But for class reduced to iter=50 because it can take a while to run

otu.rrpp <- RRPP::lm.rrpp(OTU ~ Treatment*Spp+Plot,
                   data = SAM, SS.type="III",
                   print.progress = FALSE,
                   seed="random",
                   iter=50)
```

```
##
## Warning: Because variables in the linear model are redundant,
## the linear model design has been truncated (via QR decomposition).
## Original X columns: 29
## Final X columns (rank): 28
## Check coefficients or degrees of freedom in ANOVA to see changes.
```

```
anova(otu.rrpp, effect.type = "F",
      error = c("Treatment:Spp", "Residuals", "Residuals", "Residuals"))
```

```
##
## Analysis of Variance, using Residual Randomization
## Permutation procedure: Randomization of null model residuals
## Number of permutations: 51
## Estimation method: Ordinary Least Squares
## Sums of Squares and Cross-products: Type III
## Effect sizes (Z) based on F distributions
##
##                 Df     SS     MS     Rsq      F       Z  Pr(>F)
## Treatment        1   3376 3375.9 0.00872 1.8138  3.6805 0.01961 *
## Spp             13  21174 1628.8 0.05467 0.9573 -0.8602 0.80392
## Plot             1   6802 6802.0 0.01756 3.9977  4.7717 0.01961 *
## Treatment:Spp   12  22335 1861.3 0.05766 1.0939  2.2061 0.01961 *
## Residuals      187 318178 1701.5 0.82146
## Total          214 387335
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Call: RRPP::lm.rrpp(f1 = OTU ~ Treatment * Spp + Plot, iter = 50, seed = "random",
##     SS.type = "III", data = SAM, print.progress = FALSE)
```

```
# you can run rrpp with GLS if you include a covariance matrix (Cov = )

# note that rrpp::manova.update will also provide Pillai's Trace and Roy's largest root
# but current version cannot handle mixed models (i.e., will only use MS Residual error term)
# future version will allow specification of error term - see manual
```

PermANOVA - beta diversity distance matrix

Calculate distance matrix and check assumptions

```
# get distance matrix
OTU_d <- vegan::vegdist(OTU, method="euclidean")

# test multivariate homogeneity of variances (dispersions)
# alt is to use vegan::permutest(betadisp) instead of anova
# heterogeneous variances for Treatment and Plot
# but difference is minimized compared to other distances
# and permutational approach should be robust to this
anova(vegan::betadisper(OTU_d, SAM$Treatment))
```

```
## Analysis of Variance Table
##
## Response: Distances
##            Df  Sum Sq Mean Sq F value  Pr(>F)
## Groups      1   328.6  328.55  4.7031 0.03122 *
## Residuals 213 14880.1   69.86
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
anova(vegan::betadisper(OTU_d, SAM$Spp))
```

```
## Analysis of Variance Table
##
## Response: Distances
##            Df  Sum Sq Mean Sq F value Pr(>F)
## Groups     13   726.7  55.903  0.7074 0.7549
## Residuals 201 15884.4  79.027
```

```
anova(vegan::betadisper(OTU_d, SAM$Plot))
```

```
## Analysis of Variance Table
##
## Response: Distances
##            Df Sum Sq Mean Sq F value    Pr(>F)
## Groups     12 4533.4  377.78  7.7076 1.007e-11 ***
## Residuals 202 9900.9   49.01
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

RRPP

```
# run RRPP on euclidean distance matrix - are results the same?
otu.rrpp.d <- RRPP::lm.rrpp(OTU_d ~ Treatment*Spp+Plot,
                   data = SAM, SS.type="III",
                   print.progress = FALSE,
                   seed="random",
                   iter=50)
```

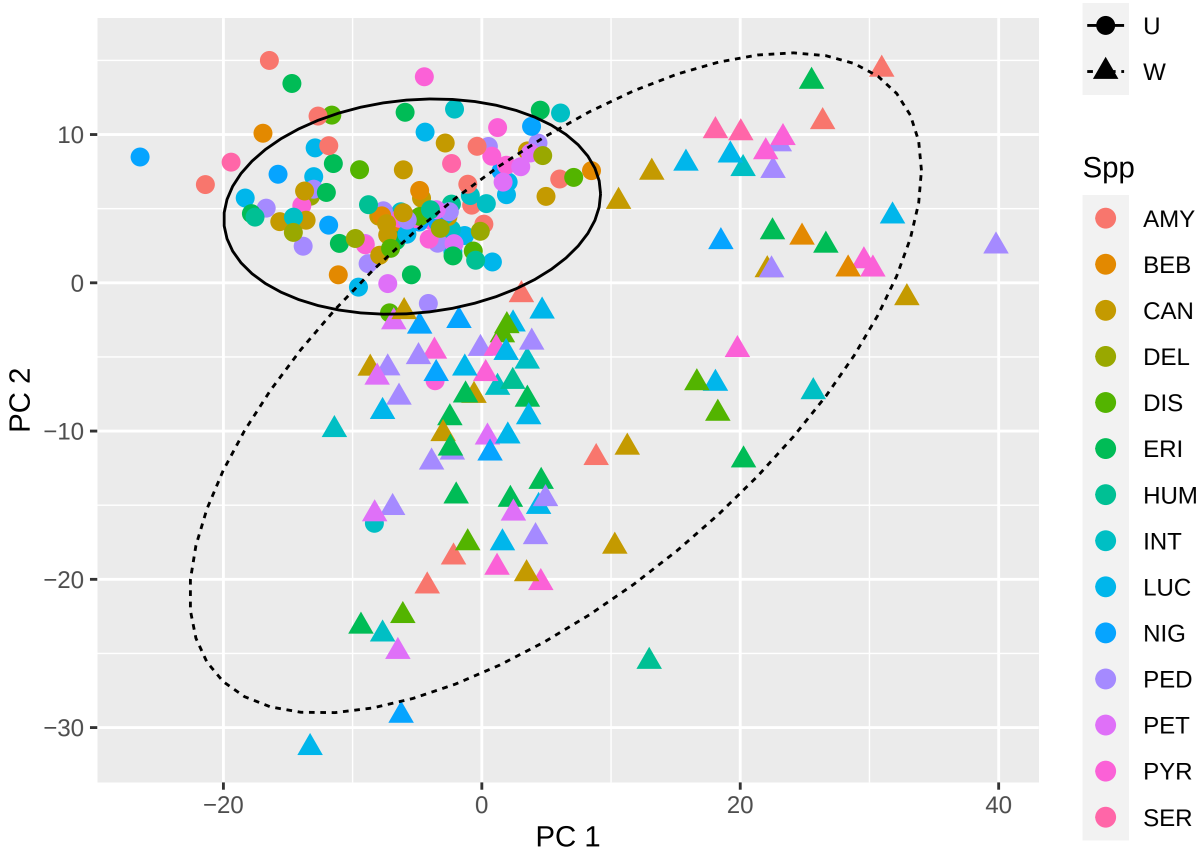
```
##
## Warning: Because variables in the linear model are redundant,
## the linear model design has been truncated (via QR decomposition).
## Original X columns: 29
## Final X columns (rank): 28
## Check coefficients or degrees of freedom in ANOVA to see changes.
```

```
anova(otu.rrpp.d, effect.type = "F",
      error = c("Treatment:Spp", "Residuals", "Residuals", "Residuals"))
```

```
##
## Analysis of Variance, using Residual Randomization
## Permutation procedure: Randomization of null model residuals
## Number of permutations: 51
## Estimation method: Ordinary Least Squares
## Sums of Squares and Cross-products: Type III
## Effect sizes (Z) based on F distributions
##
##                 Df     SS     MS     Rsq      F       Z  Pr(>F)
## Treatment        1   3376 3375.9 0.00872 1.8138  4.1669 0.01961 *
## Spp             13  21174 1628.8 0.05467 0.9573 -0.8306 0.78431
## Plot             1   6802 6802.0 0.01756 3.9977  2.7388 0.01961 *
## Treatment:Spp   12  22335 1861.3 0.05766 1.0939  1.7785 0.05882 .
## Residuals      187 318178 1701.5 0.82146
## Total          214 387335
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Call: RRPP::lm.rrpp(f1 = OTU_d ~ Treatment * Spp + Plot, iter = 50,
##     seed = "random", SS.type = "III", data = SAM, print.progress = FALSE)
```

Visualize with PCoA ord scores on vst-transformed data

```
ggplot(data = SAM, aes(x = Axis.1, y = Axis.2, color = Spp, shape = Treatment)) +
      geom_point(size = 3) + xlab("PC 1") + ylab("PC 2") +
      stat_ellipse(data=SAM, aes(x=Axis.1, y=Axis.2, lty=Treatment), inherit.aes=FALSE)
```


PC 1
PC 2
−20
0
20
40
10
0
−10
−20
−30
U
W
Spp
AMY
BEB
CAN
DEL
DIS
ERI
HUM
INT
LUC
NIG
PED
PET
PYR
SER

Coding Exercises

1. Rerun PermANOVA on non-euclidean distance matrix

- Start with the otu matrix (already vst transformed)
- Select a distance metric such as Bray Curtis, Jaccard, etc.
- Rerun RRPP - how does this compare to earlier results?
- Visualize with ordination

2. Build a new permANOVA model

- Use data from Wagner et al. 2016 - import these files from GitHub:
  - "Wk12_Wagner_SAM.csv"
  - "Wk12_Wagner_ASV.csv"
- Original data were reduced as follows:
  - limited to samples in 2011 (phyloseq::subset_samples)
  - limited to the ecotype experiment (phyloseq::subset_samples)
  - removed one site with fewer blocks (phyloseq::subset_samples)
  - removed unidentified taxa (phyloseq::subset_taxa)
  - removed taxa with less than 20 reads (phyloseq::prune_taxa)
  - if your computer is slow, consider removing taxa <50 reads
- For this coding exercise:
  - Transform the data with clr or vst (your choice)
  - Examine the experimental factors in the SAM_data file
  - Use original paper to understand fixed vs. random effects
    - https://www.nature.com/articles/ncomms12151
  - Define a simplified factorial design
    - Simplified design is to allow for faster calculation of permutations
    - Include two fixed effects
    - Include one random effect
    - Include interactions with random effects (may have to specify entire model)

    - Run a permANOVA based on your design using RRPP
        - Check that the F ratios were calculated correctly
        - Use RRPP::pairwise for posthoc tests for significant factors
            - Limit to those with >2 treatment levels
    - Visualize results with an ordination
    - Interpret the results

3. **Test number of permutations**

- Create a simplified fixed effects model from the Wagner et al. data
    - Use two fixed effects (ignore random effects for simplicity)
- Run RRPP permANOVA with increasing permutations (e.g., iter=10, 100, 1000)
- Describe how the number of permutations changes model results

Session Info

```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] ggordiplots_0.4.0           glue_1.6.2
##  [3] vegan_2.5-7                 lattice_0.20-45
##  [5] permute_0.9-7               RRPP_1.1.2
##  [7] DESeq2_1.34.0               SummarizedExperiment_1.24.0
##  [9] Biobase_2.54.0              MatrixGenerics_1.6.0
## [11] matrixStats_0.61.0          GenomicRanges_1.46.1
## [13] GenomeInfoDb_1.30.1         IRanges_2.28.0
## [15] S4Vectors_0.32.3            BiocGenerics_0.40.0
## [17] phyloseq_1.38.0             forcats_0.5.1
## [19] stringr_1.4.0               dplyr_1.0.8
## [21] purrr_0.3.4                 readr_2.1.2
## [23] tidyr_1.2.0                 tibble_3.1.6
## [25] ggplot2_3.3.5               tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##   [1] readxl_1.3.1           backports_1.4.1        plyr_1.8.6
##   [4] igraph_1.2.11          splines_4.1.2          BiocParallel_1.28.3
##   [7] TH.data_1.1-0          digest_0.6.29          foreach_1.5.2
##  [10] htmltools_0.5.2        fansi_1.0.2            magrittr_2.0.2
##  [13] memoise_2.0.1          cluster_2.1.2          tzdb_0.2.0
##  [16] Biostrings_2.62.0      annotate_1.72.0        modelr_0.1.8
##  [19] sandwich_3.0-1         colorspace_2.0-3       blob_1.2.2
##  [22] rvest_1.0.2            haven_2.4.3            xfun_0.29
##  [25] crayon_1.5.0           RCurl_1.98-1.6         jsonlite_1.8.0
##  [28] libcoin_1.0-9          genefilter_1.76.0      Exact_3.1
##  [31] zoo_1.8-9              survival_3.2-13        iterators_1.0.14
##  [34] ape_5.6-2              gtable_0.3.0           zlibbioc_1.40.0
##  [37] XVector_0.34.0         DelayedArray_0.20.0    Rhdf5lib_1.16.0
##  [40] scales_1.1.1           mvtnorm_1.1-3          DBI_1.1.2
##  [43] Rcpp_1.0.8.3           xtable_1.8-4           bit_4.0.4
##  [46] proxy_0.4-26           rcompanion_2.4.13      httr_1.4.2
##  [49] RColorBrewer_1.1-2     ellipsis_0.3.2         modeltools_0.2-23
##  [52] farver_2.1.0           pkgconfig_2.0.3        XML_3.99-0.9
##  [55] multcompView_0.1-8     dbplyr_2.1.1           locfit_1.5-9.4
##  [58] utf8_1.2.2             labeling_0.4.2         tidyselect_1.1.2
##  [61] rlang_1.0.2            reshape2_1.4.4         AnnotationDbi_1.56.2
##  [64] munsell_0.5.0          cellranger_1.1.0       tools_4.1.2
##  [67] cachem_1.0.6           cli_3.2.0              generics_0.1.2
##  [70] RSQLite_2.2.10         ade4_1.7-18            broom_0.7.12
##  [73] evaluate_0.15          biomformat_1.22.0      fastmap_1.1.0
##  [76] yaml_2.3.5             knitr_1.37             bit64_4.0.5
##  [79] fs_1.5.2               KEGGREST_1.34.0        coin_1.4-2
##  [82] rootSolve_1.8.2.3      nlme_3.1-155           xml2_1.3.3
##  [85] compiler_4.1.2         rstudioapi_0.13        png_0.1-7
##  [88] e1071_1.7-9            reprex_2.0.1           geneplotter_1.72.0
##  [91] DescTools_0.99.44      stringi_1.7.6          highr_0.9
##  [94] Matrix_1.4-0           multtest_2.50.0        vctrs_0.3.8
##  [97] pillar_1.7.0           lifecycle_1.0.1        rhdf5filters_1.6.0
## [100] lmtest_0.9-39          data.table_1.14.2      bitops_1.0-7
## [103] lmom_2.8               R6_2.5.1               gld_2.6.4
## [106] codetools_0.2-18       boot_1.3-28            MASS_7.3-54
## [109] assertthat_0.2.1       rhdf5_2.38.0           nortest_1.0-4
## [112] withr_2.5.0            multcomp_1.4-18        GenomeInfoDbData_1.2.7
## [115] mgcv_1.8-39            expm_0.999-6           parallel_4.1.2
## [118] hms_1.1.1              grid_4.1.2             class_7.3-19
## [121] rmarkdown_2.11         lubridate_1.8.0
```