

Beta diversity: distance-based Redundancy Analysis

Stijn Schreven

23 April 2022

Contents

Introduction	2
Load packages	2
Input files	2
1. Prepare data	3
1.1. Creating balanced subsets	3
1.2. Data format for dbRDA	4
1.3. Function for pairwise PERMANOVA contrasts	5
2. dbRDA: substrates	5
2.1. Total	5
2.2. Chicken feed	11
2.3. Camelina	13
2.4. Chicken manure	15
3. dbRDA: larvae and substrates	17
3.1. Total	18
3.2. Chicken feed	20
3.3. Camelina	23
3.4. Chicken manure	25
4. dbRDA: larvae	27
4.1. Total	27
4.2. Chicken feed	29
4.3. Camelina	32
4.4. Chicken manure	34

Introduction

We used weighted UniFrac distance-based Redundancy Analysis (dbRDA) as constrained ordination to investigate how much microbiota variation was explained by our treatments. We conducted dbRDA by adapting the R code of Shankar *et al.* 2017 (*i.e.* the direct use of the distance matrix from **phyloseq::distance(pseq, “wunifrac”)**) as input in the ordination function). Analysis done at genus level, relative abundance.

Important difference from Shankar *et al.* 2017:

- We used direct dbRDA (**vegan::dbrda**, McArdle & Anderson 2001) partitioning the distance matrix, instead of CAP (**vegan::capscale** or **phyloseq::ordinate(method = “CAP”)**, Legendre & Anderson 1999) based on PCoA which includes negative eigenvalues and imaginary axes. In the direct dbRDA, negative eigenvalues occur, but no correction is needed.

Input files for dbRDA:

- dataframe of response variables (transposed `otu_table`: rows = samples, columns = variables). This is optional, only needed to obtain species scores;
- dataframe of explanatory variables (metadata);
- distance matrix of weighted UniFrac distances.

Settings for PERMANOVA **anova.cca**:

- stratifying for repeated measures of container ID via the argument **permutations = how(within = Within(type = “none”), plots = Plots(strata = env\$ContainerID, type = “free”), nperm = 999)**. This restricts permutations to the level of plots (containers), but observations within plots are not permuted;
- **N.B.:** the input has to be a balanced dataset for correct permutations. This means that all levels of factor interactions should be present, *i.e.* crossed design, and all groups within the plot stratum should have equal sample sizes.

For more info on random effects and % inertia explained: <https://cran.r-project.org/web/packages/vegan/vignettes/FAQ-vegan.html>

And for calculation of % inertia explained per treatment: <http://edild.github.io/prc1/> and <http://edild.github.io/prc2/>

Assess goodness of fit by **stressplot.dbrda**.

Load packages

```
library(phyloseq)
library(microbiome)
library(vegan)
```

Input files

```
pstot <- readRDS("./phyobjects/ps1.exp.rds")

# transform to genus level, relative abundance
pstot.g <- aggregate_taxa(pstot, "Genus")
pstot.g.r <- microbiome::transform(pstot.g, "compositional")
```

1. Prepare data

1.1. Creating balanced subsets

```
# all diets, substrates, excl. timepoint 0, excl. larvae
pstot.s <- subset_samples(pstot.g.r, Type == "substrate"& Timepoint != 0)
pstot.s <- prune_taxa(taxa_sums(otu_table(pstot.s)) > 0, pstot.s)

# all diets, substrates & larvae, excl. timepoint 0, excl. density 0
pstot.ls <- subset_samples(pstot.g.r, Timepoint != 0 & Density != 0)
pstot.ls <- prune_taxa(taxa_sums(otu_table(pstot.ls)) > 0, pstot.ls)

# all diets, larvae
pstot.l <- subset_samples(pstot.g.r, Type == "larvae")
pstot.l <- prune_taxa(taxa_sums(otu_table(pstot.l)) > 0, pstot.l)

# per diet, substrates, incl. timepoint 0 (for CF and CM), excl. larvae
CFs <- subset_samples(pstot.g.r, Diet == "CF" & Type == "substrate")
CFs <- prune_taxa(taxa_sums(otu_table(CFs)) > 0, CFs)
CSs <- subset_samples(pstot.g.r, Diet == "CS" & Type == "substrate")
CSs <- prune_taxa(taxa_sums(otu_table(CSs)) > 0, CSs)
CMs <- subset_samples(pstot.g.r, Diet == "CM" & Type == "substrate")
CMs <- prune_taxa(taxa_sums(otu_table(CMs)) > 0, CMs)

# per diet, substrates & larvae, excl. timepoint 0, excl. density 0
CF.ls <- subset_samples(pstot.ls, Diet == "CF")
CF.ls <- prune_taxa(taxa_sums(otu_table(CF.ls)) > 0, CF.ls)
CS.ls <- subset_samples(pstot.ls, Diet == "CS")
CS.ls <- prune_taxa(taxa_sums(otu_table(CS.ls)) > 0, CS.ls)
CM.ls <- subset_samples(pstot.ls, Diet == "CM")
CM.ls <- prune_taxa(taxa_sums(otu_table(CM.ls)) > 0, CM.ls)

# per diet, larvae
CF1 <- subset_samples(pstot.g.r, Diet == "CF" & Type == "larvae")
CF1 <- prune_taxa(taxa_sums(otu_table(CF1)) > 0, CF1)
CS1 <- subset_samples(pstot.g.r, Diet == "CS" & Type == "larvae")
CS1 <- prune_taxa(taxa_sums(otu_table(CS1)) > 0, CS1)
CM1 <- subset_samples(pstot.g.r, Diet == "CM" & Type == "larvae")
CM1 <- prune_taxa(taxa_sums(otu_table(CM1)) > 0, CM1)
```

1.2. Data format for dbRDA

```
# substrates

resp.sub <- as.data.frame(t(abundances(pstot.s))) # response data
#                                                  |_ optional for species scores
env.sub  <- meta(pstot.s)                        # metadata
wuf.sub  <- distance(pstot.s, "wunifrac")        # distance matrix

env.cfs  <- meta(CFs)
wuf.cfs  <- distance(CFs, "wunifrac")

env.cms  <- meta(CMs)
wuf.cms  <- distance(CMs, "wunifrac")

env.css  <- meta(CSs)
wuf.css  <- distance(CSs, "wunifrac")

# larvae and substrates

env.ls   <- meta(pstot.ls)
wuf.ls   <- distance(pstot.ls, "wunifrac")

env.cf    <- meta(CF.ls)
wuf.cf    <- distance(CF.ls, "wunifrac")

env.cm    <- meta(CM.ls)
wuf.cm    <- distance(CM.ls, "wunifrac")

env.cs    <- meta(CS.ls)
wuf.cs    <- distance(CS.ls, "wunifrac")

# larvae

resp.l <- as.data.frame(t(abundances(pstot.l)))
env.l  <- meta(pstot.l)
wuf.l  <- distance(pstot.l, "wunifrac")

env.cfl <- meta(CFl)
wuf.cfl <- distance(CFl, "wunifrac")

env.csl <- meta(CSl)
wuf.csl <- distance(CSl, "wunifrac")

env.cml <- meta(CMl)
wuf.cml <- distance(CMl, "wunifrac")
```

1.3. Function for pairwise PERMANOVA contrasts

Pairwise contrasts between groups of a categorical variable. R code adapted from ResearchGate, with the following changes:

- using weighted UniFrac distance object as input;
- restricting permutations by strata of containerID using plots argument;
- function can be created using either **anova.cca** or **adonis**; the output is identical. Here, we chose to use **anova.cca**.

Source code: https://www.researchgate.net/post/How_can_I_do_PerMANOVA_pairwise_contrasts_in_R

```
pairwise.aov.cca <- function(x, meta, factors, p.adjust.m = 'fdr')
{
  # x = distance object (phyloseq::distance(pseq, "wunifrac") output)
  # meta = metadata (phyloseq::meta(pseq))
  # factors = a column in metadata specifying the factor for comparisons
  # N.B.: metadata should have a column "ContainerID" for subject ID for strata
  co = combn(unique(as.character(factors)),2)
  pairs = c()
  F.Model = c()
  R2 = c()
  p.value = c()
  for(elem in 1:ncol(co)){
    x1 = as.matrix(x)[factors %in% c(co[1,elem],co[2,elem]),
                        factors %in% c(co[1,elem],co[2,elem])];
    x1.meta = meta[factors %in% c(co[1,elem],co[2,elem]),];
    x1.dbrda = dbrda(x1 ~ factors[factors %in% c(co[1,elem],co[2,elem])]);
    x1.aov = anova.cca(x1.dbrda, model = "direct", by = "terms",
                      permutations = how(within = Within(type = "none"),
                      plots = Plots(strata = x1.meta$ContainerID,
                      type = "free"), nperm = 999));
    pairs = c(pairs, paste(co[1,elem], 'vs', co[2,elem]));
    F.Model = c(F.Model, x1.aov[1,3]);
    R2 = c(R2, x1.aov[1,2]/(x1.aov[1,2] + x1.aov[2,2]));
    p.value = c(p.value, x1.aov[1,4])
  }
  p.adjusted = p.adjust(p.value, method = p.adjust.m)
  pairw.res = data.frame(pairs, F.Model, R2, p.value, p.adjusted)
  return(pairw.res)
}
```

2. dbRDA: substrates

Supplementary Tables S5 (total) and S6 (diets separately) in the manuscript Chapter 3 in PhD thesis. Supplementary Tables S3 and S4 in the manuscript submitted to *Applied and Environmental Microbiology*.

2.1. Total

```
# calculate dbRDA
dbRDA.wuf <- dbrda(wuf.sub ~ env.sub$Diet * env.sub$Density * env.sub$Timepoint)

dbRDA.wuf
```

```
## Call: dbrda(formula = wuf.sub ~ env.sub$Diet * env.sub$Density *
## env.sub$Timepoint)
##
##              Inertia Proportion Rank RealDims
## Total          22.2034      1.0000
## Constrained    16.6392      0.7494   35      23
## Unconstrained   5.5642      0.2506  108      52
## Inertia is squared Unknown distance
##
## Eigenvalues for constrained axes:
##   dbRDA1  dbRDA2  dbRDA3  dbRDA4  dbRDA5  dbRDA6  dbRDA7  dbRDA8
##   9.875   2.660   0.984   0.905   0.611   0.507   0.369   0.278
##   dbRDA9  dbRDA10 dbRDA11 dbRDA12 dbRDA13 dbRDA14 dbRDA15 dbRDA16
##   0.154   0.117   0.101   0.084   0.065   0.053   0.046   0.039
##   dbRDA17 dbRDA18 dbRDA19 dbRDA20 dbRDA21 dbRDA22 dbRDA23 idbRDA1
##   0.034   0.027   0.017   0.015   0.009   0.005   0.004  -0.002
##   idbRDA2 idbRDA3 idbRDA4 idbRDA5 idbRDA6 idbRDA7 idbRDA8 idbRDA9
##  -0.004  -0.007  -0.009  -0.012  -0.016  -0.019  -0.025  -0.033
##   idbRDA10 idbRDA11 idbRDA12
##  -0.046  -0.066  -0.082
##
## Eigenvalues for unconstrained axes:
##   MDS1  MDS2  MDS3  MDS4  MDS5  MDS6  MDS7  MDS8
##  2.0162 0.7575 0.6847 0.5120 0.4586 0.3709 0.3435 0.2482
## (Showing 8 of 108 unconstrained eigenvalues)
```

```
RsquareAdj(dbRDA.wuf)
```

```
## $r.squared
## [1] 0.7493978
##
## $adj.r.squared
## [1] 0.6681842
```

```
# permutation tests
dbRDA.wuf0 <- dbrda(wuf.sub ~ 1)
anova.cca(dbRDA.wuf0, dbRDA.wuf,
           permutations = how(within = Within(type = "none"),
                              plots = Plots(strata = env.sub$ContainerID,
                                             type = "free"), nperm = 999))
```

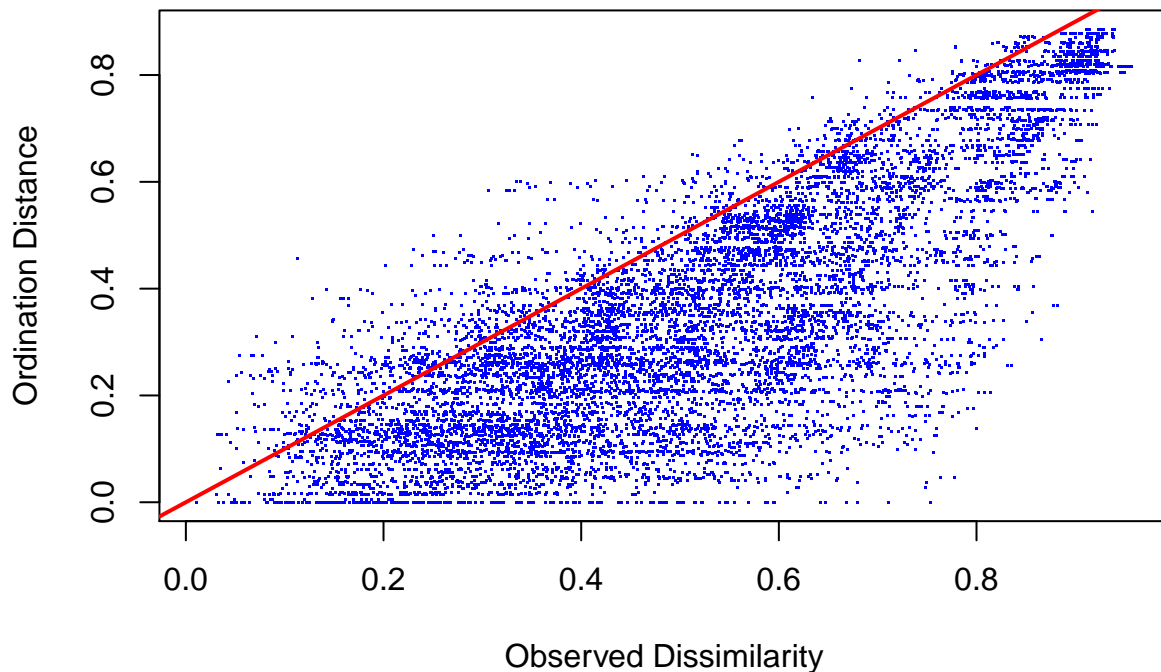
```
## Permutation tests for dbrda under reduced model
## Plots: env.sub$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
```

```
## Model 1: wuf.sub ~ 1
## Model 2: wuf.sub ~ env.sub$Diet * env.sub$Density * env.sub$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1    143      22.2034      16.639 9.2275 0.001 ***
## 2    108      5.5642 35      16.639 9.2275 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# by terms
aov.wuf <- anova.cca(dbrDA.wuf, model = "direct", by="terms",
  permutations = how(within = Within(type = "none"),
    plots = Plots(strata = env.sub$ContainerID,
      type = "free"), nperm = 999))
aov.wuf
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.sub$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.sub ~ env.sub$Diet * env.sub$Density * env.sub$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.sub$Diet    2   8.4854 82.3496 0.001 ***
## env.sub$Density  3   0.9850  6.3730 0.004 **
## env.sub$Timepoint 2   1.4393 13.9686 0.001 ***
## env.sub$Diet:env.sub$Density  6   1.1579  3.7459 0.021 *
## env.sub$Diet:env.sub$Timepoint  4   2.4472 11.8747 0.001 ***
## env.sub$Density:env.sub$Timepoint  6   0.6395  2.0687 0.001 ***
## env.sub$Diet:env.sub$Density:env.sub$Timepoint 12   1.4848  2.4017 0.001 ***
## Residual      108   5.5642
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.wuf) # good fit.
```



```
# dispersion test
anova(betadisper(wuf.sub, group = interaction(env.sub$Diet,
      env.sub$Density, env.sub$Timepoint, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.sub$ContainerID,
          type = "free"), nperm = 999))

## Warning in betadisper(wuf.sub, group = interaction(env.sub$Diet,
## env.sub$Density, : some squared distances are negative and changed to zero

## Analysis of Variance Table
##
## Response: Distances
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Groups      35  0.76506  0.021859   2.0544 0.002572 **
## Residuals  108  1.14915  0.010640
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# significant differences in dispersion: caution with anova.cca output.
```

2.1.1. Pairwise contrasts of diets


```
# pairwise contrasts of Diet
pairwise.aov.cca(x = wuf.sub, meta = env.sub, factors = env.sub$Diet)
```

```
##      pairs    F.Model      R2 p.value p.adjusted
## 1 CF vs CS  7.764276 0.07629667  0.001      0.001
## 2 CF vs CM 53.790924 0.36396636  0.001      0.001
## 3 CS vs CM 69.933897 0.42659815  0.001      0.001
```

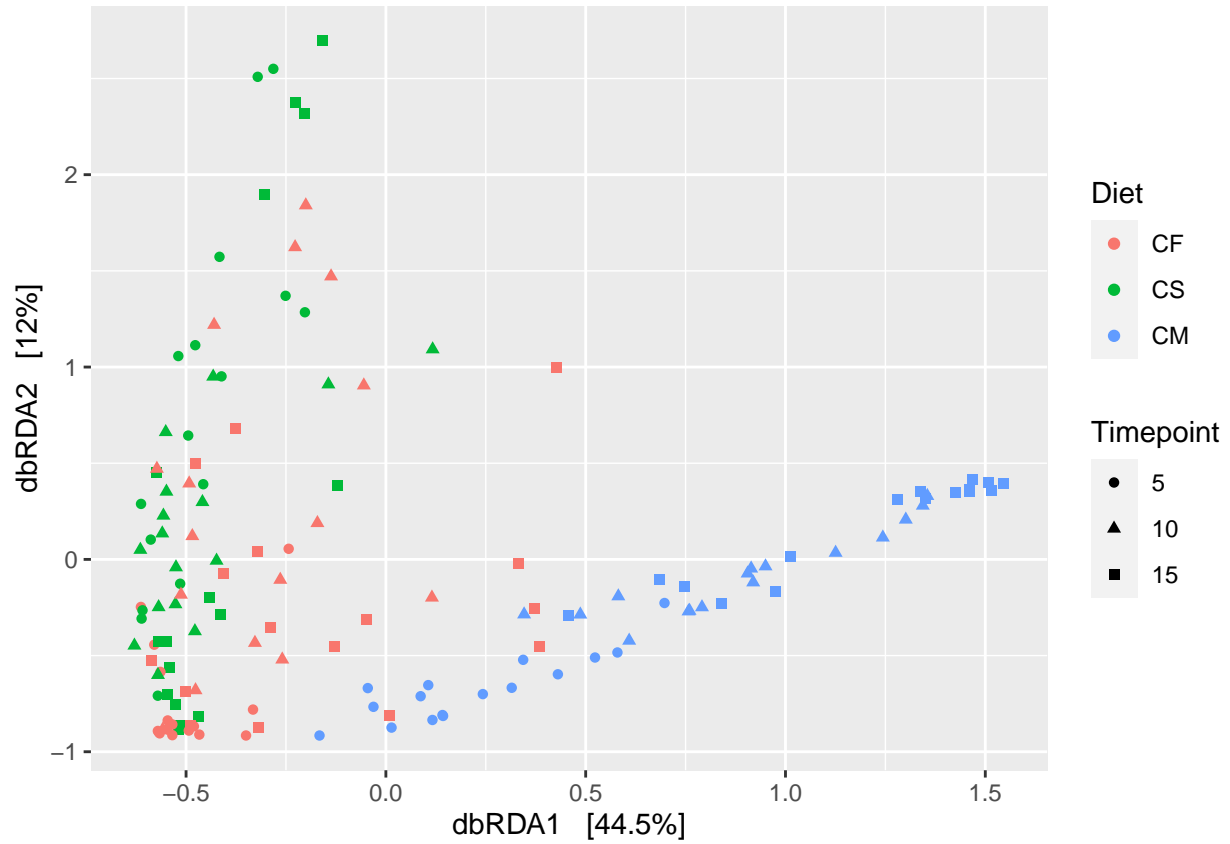
```
# result: substrate microbiota of all diets differ from each other.
```

```
# it is also possible to test pairwise contrasts of interactions, here Diet x Timepoint
pairwise.aov.cca(x = wuf.sub, meta = env.sub, factors =
  interaction(env.sub$Diet, env.sub$Timepoint, drop = T))
```

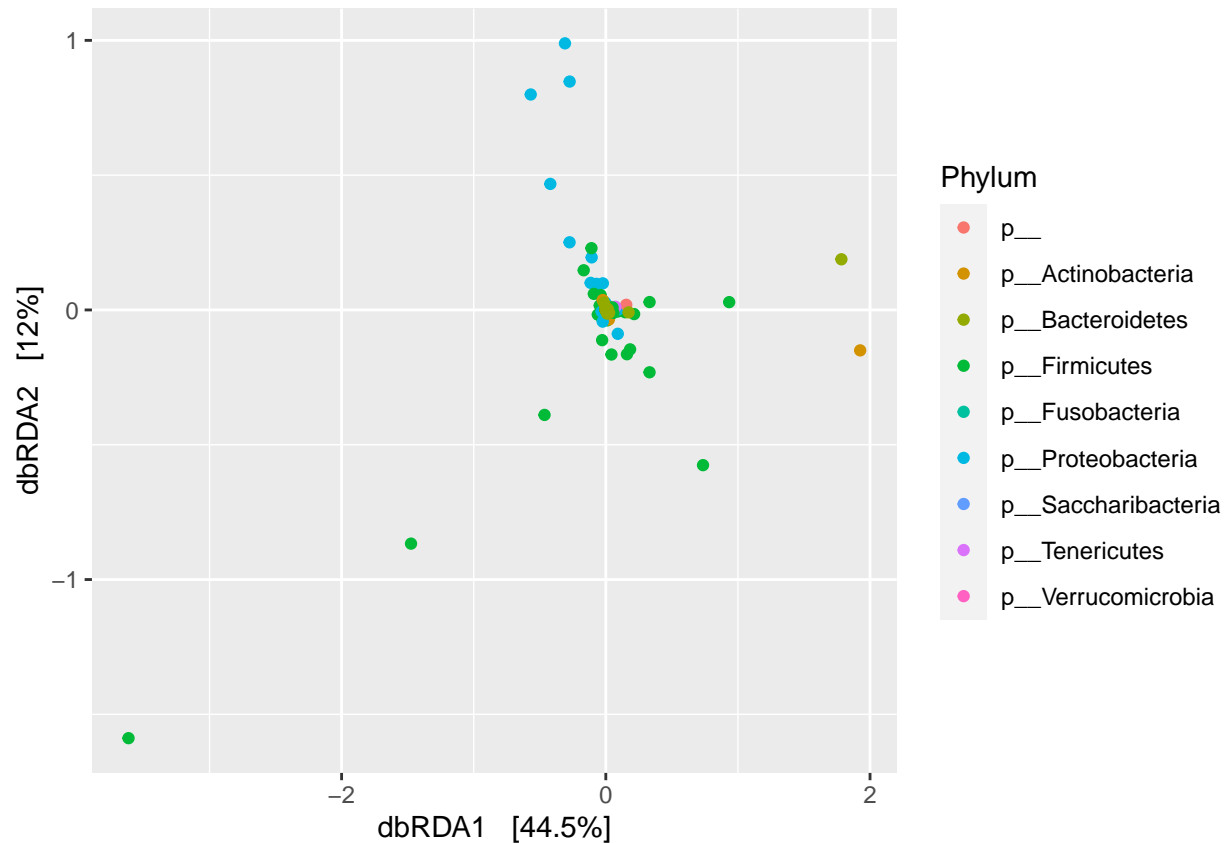
```
##      pairs    F.Model      R2 p.value p.adjusted
## 1  CF.5 vs CF.10 13.052894 0.30318272  0.001 0.001241379
## 2  CF.5 vs CS.10 15.225085 0.33665133  0.001 0.001241379
## 3  CF.5 vs CM.10 77.249254 0.72027777  0.001 0.001241379
## 4  CF.5 vs CS.15  7.654854 0.20328997  0.001 0.001241379
## 5  CF.5 vs CF.15 10.798592 0.26468050  0.001 0.001241379
## 6  CF.5 vs CM.15 90.846812 0.75175183  0.001 0.001241379
## 7   CF.5 vs CM.5 44.280366 0.59612477  0.001 0.001241379
## 8   CF.5 vs CS.5 21.573980 0.41831132  0.001 0.001241379
## 9  CF.10 vs CS.10  3.649779 0.10846368  0.005 0.006000000
## 10 CF.10 vs CM.10 30.201972 0.50167745  0.001 0.001241379
## 11 CF.10 vs CS.15  2.004972 0.06264564  0.108 0.108000000
## 12 CF.10 vs CF.15  3.720975 0.11034602  0.001 0.001241379
## 13 CF.10 vs CM.15 40.996803 0.57744576  0.001 0.001241379
## 14  CF.10 vs CM.5 16.010661 0.34797720  0.001 0.001241379
## 15  CF.10 vs CS.5  3.665269 0.10887390  0.018 0.019636364
## 16 CS.10 vs CM.10 48.499888 0.61783385  0.001 0.001241379
## 17 CS.10 vs CS.15  2.676342 0.08190459  0.093 0.095657143
## 18 CS.10 vs CF.15  6.487601 0.17780290  0.001 0.001241379
## 19 CS.10 vs CM.15 61.574726 0.67239869  0.001 0.001241379
## 20 CS.10 vs CM.5 25.161682 0.45614421  0.001 0.001241379
## 21 CS.10 vs CS.5  2.879839 0.08758678  0.014 0.015750000
## 22 CM.10 vs CS.15 27.957615 0.48238036  0.001 0.001241379
## 23 CM.10 vs CF.15 21.684325 0.41955323  0.001 0.001241379
## 24 CM.10 vs CM.15  5.144404 0.14637905  0.001 0.001241379
## 25  CM.10 vs CM.5 19.778494 0.39733010  0.001 0.001241379
## 26  CM.10 vs CS.5 44.401985 0.59678495  0.001 0.001241379
## 27 CS.15 vs CF.15  3.829115 0.11318992  0.010 0.011612903
## 28 CS.15 vs CM.15 36.642422 0.54983629  0.001 0.001241379
## 29 CS.15 vs CM.5 13.996676 0.31813030  0.001 0.001241379
## 30 CS.15 vs CS.5  3.557791 0.10601981  0.026 0.027529412
## 31 CF.15 vs CM.15 31.873286 0.51513809  0.001 0.001241379
## 32 CF.15 vs CM.5  8.512912 0.22104046  0.001 0.001241379
## 33 CF.15 vs CS.5  9.031644 0.23139286  0.001 0.001241379
## 34 CM.15 vs CM.5 37.465857 0.55533063  0.001 0.001241379
## 35 CM.15 vs CS.5 55.367476 0.64857811  0.001 0.001241379
## 36  CM.5 vs CS.5 30.464075 0.50383761  0.001 0.001241379
```

2.1.2. Plot dbRDA

```
# plot samples
plot_ordination(pstot.s, dbRDA.wuf, type = "samples", color = "Diet", shape = "Timepoint")
```



```
# plot taxa
sppscores(dbRDA.wuf) <- resp.sub
plot_ordination(pstot.s, dbRDA.wuf, type = "taxa", color = "Phylum")
```



2.2. Chicken feed

```
# Calculate dbRDA
dbRDA.cf.wuf <- dbrda(wuf.cfs ~ env.cfs$Density * env.cfs$Timepoint)
RsquareAdj(dbRDA.cf.wuf)
```

```
## $r.squared
## [1] 0.7859249
##
## $adj.r.squared
## [1] 0.7190264
```

```
# permutation tests
dbRDA.cf.wuf0 <- dbrda(wuf.cfs ~ 1)
anova(dbRDA.cf.wuf0, dbRDA.cf.wuf,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.cfs$ContainerID,
                                           type = "free"), nperm = 999))
```

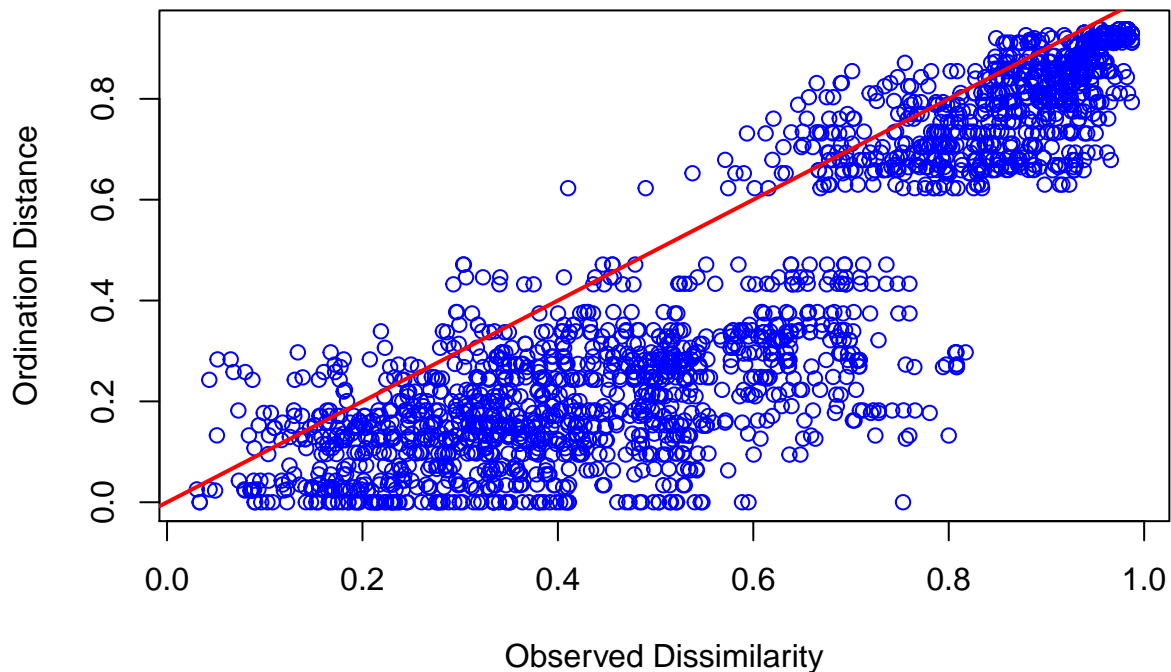
```
## Permutation tests for dbrda under reduced model
## Plots: env.cfs$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
```

```
##
## Model 1: wuf.cfs ~ 1
## Model 2: wuf.cfs ~ env.cfs$Density * env.cfs$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     63      12.7327
## 2     48       2.7258 15    10.007 11.748 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# by terms
aov.cf.wuf <- anova.cca(dbrDA.cf.wuf, model = "direct", by="terms",
  permutations = how(within = Within(type = "none"),
    plots = Plots(strata = env.cfs$ContainerID,
      type = "free"), nperm = 999))
aov.cf.wuf

## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cfs$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.cfs ~ env.cfs$Density * env.cfs$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.cfs$Density    3   0.5672  3.3297 0.001 ***
## env.cfs$Timepoint    3   8.5711 50.3118 0.001 ***
## env.cfs$Density:env.cfs$Timepoint  9   0.8686  1.6996 0.142
## Residual          48   2.7258
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# goodness of fit
stressplot(dbrDA.cf.wuf) # fit OK.
```



```
# dispersion test
anova(betadisper(wuf.cfs, group = interaction(env.cfs$Density,
      env.cfs$Timepoint, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.cfs$ContainerID,
          type = "free"), nperm = 999))

## Analysis of Variance Table
##
## Response: Distances
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups  15 0.28407  0.0189381   2.5223 0.007773 **
## Residuals 48 0.36039  0.0075081
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

2.3. Camelina

```
# Calculate dbRDA
dbRDA.cs.wuf <- dbrda(wuf.css ~ env.css$Density * env.css$Timepoint)
RsquareAdj(dbRDA.cs.wuf)
```

```
## $r.squared
## [1] 0.4837163
##
## $adj.r.squared
## [1] 0.325963
```

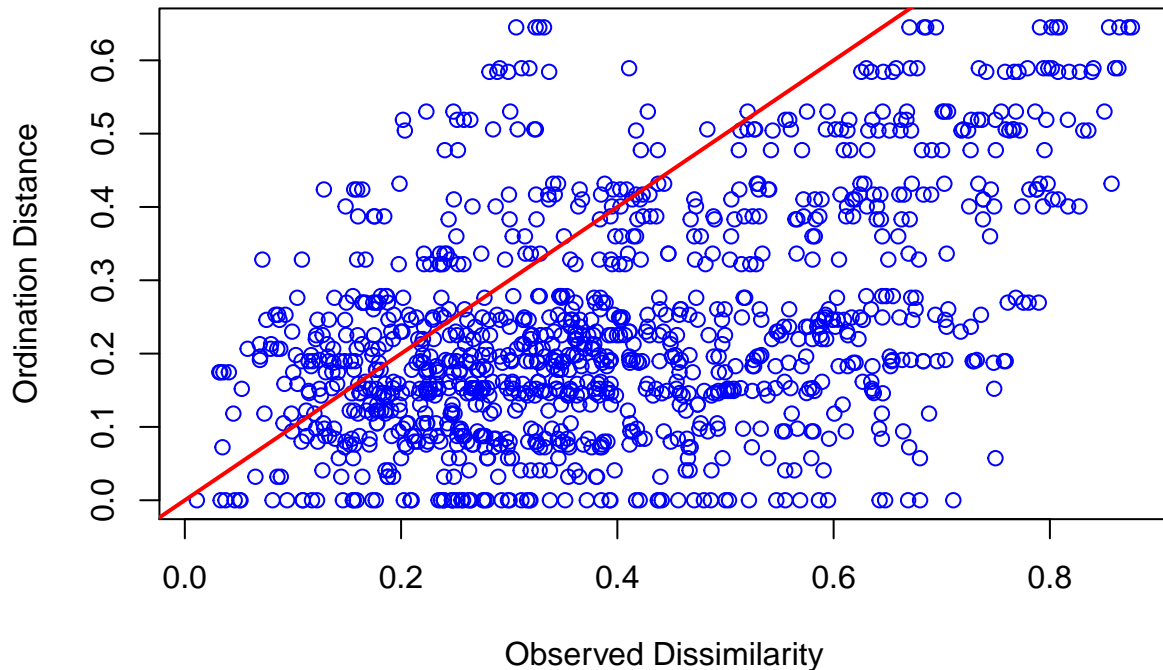
```
# permutation tests
dbRDA.cs.wuf0 <- dbrda(wuf.css ~ 1)
anova(dbRDA.cs.wuf0, dbRDA.cs.wuf,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.css$ContainerID,
                                           type = "free"), nperm = 999))
```

```
## Permutation tests for dbrda under reduced model
## Plots: env.css$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.css ~ 1
## Model 2: wuf.css ~ env.css$Density * env.css$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     47         4.5445
## 2     36         2.3463 11     2.1983 3.0663 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# by terms
aov.cs.wuf <- anova.cca(dbRDA.cs.wuf, model = "direct", by="terms",
                        permutations = how(within = Within(type = "none"),
                                           plots = Plots(strata = env.css$ContainerID,
                                                           type = "free"), nperm = 999))
aov.cs.wuf
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.css$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.css ~ env.css$Density * env.css$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.css$Density    3  0.67489 3.4518 0.002 **
## env.css$Timepoint  2  0.54607 4.1893 0.002 **
## env.css$Density:env.css$Timepoint  6  0.97730 2.4992 0.003 **
## Residual          36  2.34626
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbRDA.cs.wuf) # fit OK.
```



```
# dispersion test
anova(betadisper(wuf.css, group = interaction(env.css$Density,
env.css$Timepoint, drop = T)),
permutations = how(within = Within(type = "none"),
plots = Plots(strata = env.css$ContainerID,
type = "free"), nperm = 999))

## Warning in betadisper(wuf.css, group = interaction(env.css$Density,
## env.css$Timepoint, : some squared distances are negative and changed to zero

## Analysis of Variance Table
##
## Response: Distances
##          Df Sum Sq Mean Sq F value Pr(>F)
## Groups    11 0.27119  0.024654   1.4337 0.2005
## Residuals  36 0.61904  0.017195

# no significant difference in dispersion.
```

2.4. Chicken manure

```
# Calculate dbRDA
dbRDA.cm.wuf <- dbRDA(wuf.cms ~ env.cms$Density * env.cms$Timepoint)
RsquareAdj(dbRDA.cm.wuf)
```

```
## $r.squared
## [1] 0.8371899
##
## $adj.r.squared
## [1] 0.7863118
```

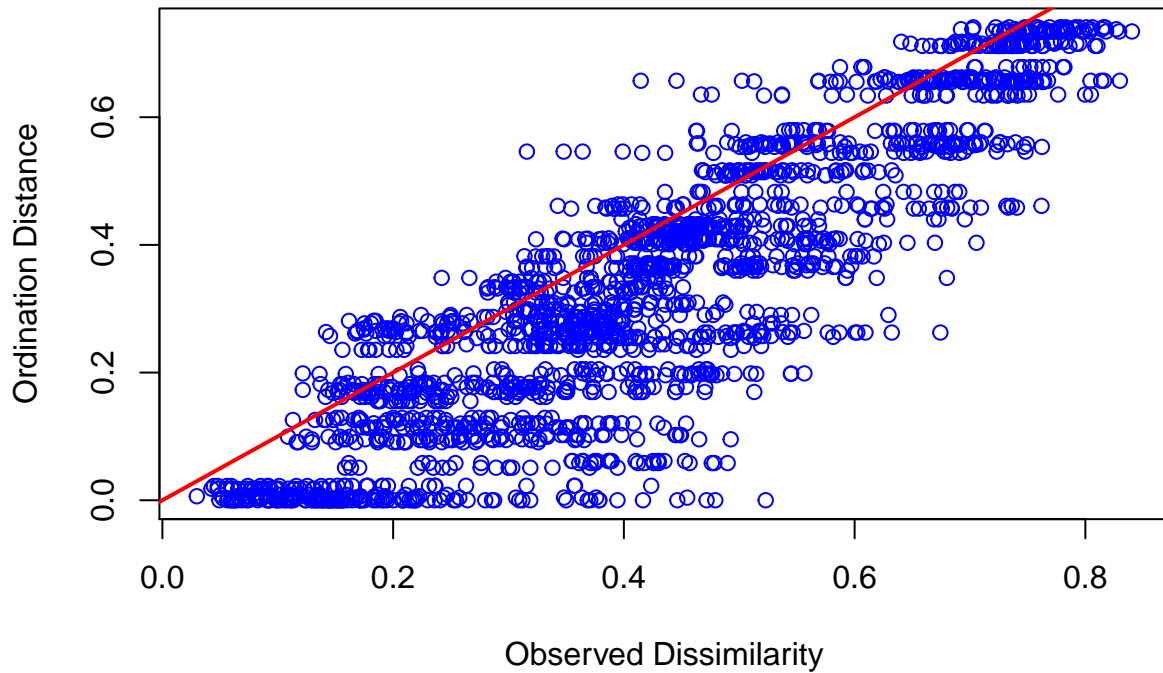
```
# permutation tests
dbRDA.cm.wuf0 <- dbrda(wuf.cms ~ 1)
anova(dbRDA.cm.wuf0, dbRDA.cm.wuf,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.cms$ContainerID,
                                           type = "free"), nperm = 999))
```

```
## Permutation tests for dbrda under reduced model
## Plots: env.cms$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.cms ~ 1
## Model 2: wuf.cms ~ env.cms$Density * env.cms$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     63          6.7279
## 2     48          1.0954 15      5.6325 16.455 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.cm.wuf <- anova.cca(dbRDA.cm.wuf, model = "direct", by="terms",
                        permutations = how(within = Within(type = "none"),
                                           plots = Plots(strata = env.cms$ContainerID,
                                                           type = "free"), nperm = 999))
aov.cm.wuf
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cms$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.cms ~ env.cms$Density * env.cms$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.cms$Density    3  0.6953 10.156 0.001 ***
## env.cms$Timepoint  3  4.2231 61.687 0.001 ***
## env.cms$Density:env.cms$Timepoint  9  0.7141  3.477 0.001 ***
## Residual          48  1.0954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbRDA.cm.wuf) # fit good.
```

```
# dispersion test
anova(betadisper(wuf.cms, group = interaction(env.cms$Density,
      env.cms$Timepoint, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.cms$ContainerID,
          type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.cms, group = interaction(env.cms$Density,
## env.cms$Timepoint, : some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
##
## Response: Distances
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups  15 0.11720  0.0078134   1.2969 0.2411
## Residuals 48 0.28919  0.0060248
```

```
# no significant differences in dispersion.
```

3. dbRDA: larvae and substrates

Supplementary Tables S7 (total) and S8 (diets separately) in the manuscript Chapter 3 in PhD thesis. Not included in the manuscript submitted to *Applied and Environmental Microbiology*.

3.1. Total

```
# calculate dbRDA
dbRDA.ls <- dbrda(wuf.ls ~ env.ls$Diet * env.ls$Density * env.ls$Timepoint * env.ls$Type)
RsquareAdj(dbRDA.ls)

## $r.squared
## [1] 0.7460044
##
## $adj.r.squared
## [1] 0.6629071

# permutation tests
dbRDA.ls0 <- dbrda(wuf.ls ~ 1)
anova.cca(dbRDA.ls0, dbRDA.ls,
           permutations = how(within = Within(type = "none"),
                              plots = Plots(strata = env.ls$ContainerID,
                                             type = "free"), nperm = 999))

## Permutation tests for dbrda under reduced model
## Plots: env.ls$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.ls ~ 1
## Model 2: wuf.ls ~ env.ls$Diet * env.ls$Density * env.ls$Timepoint * env.ls$Type
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1    215          37.307
## 2    162           9.476 53    27.831 8.9775 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# by terms
aov.ls <- anova.cca(dbRDA.ls, model = "direct", by="terms",
                   permutations = how(within = Within(type = "none"),
                                      plots = Plots(strata = env.ls$ContainerID,
                                                     type = "free"), nperm = 999))
aov.ls

## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.ls$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.ls ~ env.ls$Diet * env.ls$Density * env.ls$Timepoint * env.ls$Type)
##
##               Df SumOfSqs      F
## env.ls$Diet    2  12.8380 109.7397
## env.ls$Density  2   0.4754   4.0634
## env.ls$Timepoint 2   2.1929  18.7449
## env.ls$Type    1   1.1173  19.1020
```

```

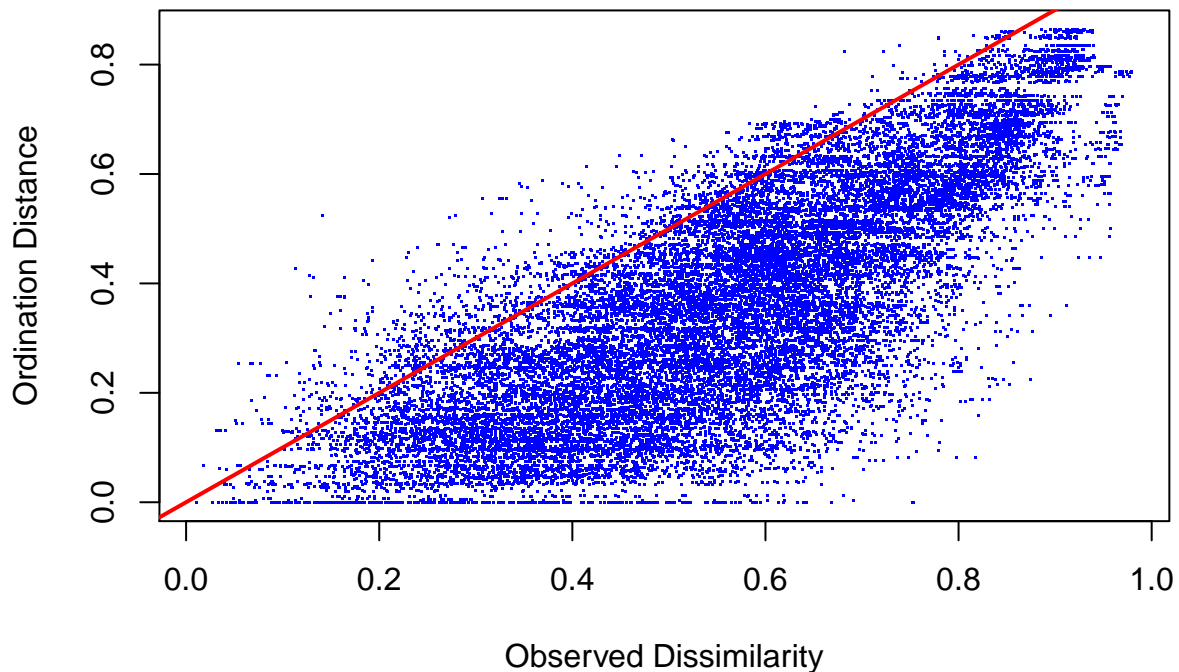
## env.ls$Diet:env.ls$Density          4  0.8797  3.7600
## env.ls$Diet:env.ls$Timepoint         4  4.8642 20.7897
## env.ls$Density:env.ls$Timepoint      4  0.6365  2.7203
## env.ls$Diet:env.ls$Type              2  0.7511  6.4206
## env.ls$Density:env.ls$Type           2  0.1982  1.6944
## env.ls$Timepoint:env.ls$Type         2  0.4653  3.9777
## env.ls$Diet:env.ls$Density:env.ls$Timepoint 8  1.6707  3.5704
## env.ls$Diet:env.ls$Density:env.ls$Type      4  0.2402  1.0268
## env.ls$Diet:env.ls$Timepoint:env.ls$Type    4  0.8180  3.4960
## env.ls$Density:env.ls$Timepoint:env.ls$Type  4  0.2794  1.1943
## env.ls$Diet:env.ls$Density:env.ls$Timepoint:env.ls$Type 8  0.4043  0.8640
## Residual                             162  9.4759
##                                     Pr(>F)
## env.ls$Diet                        0.001 ***
## env.ls$Density                     0.205
## env.ls$Timepoint                   0.001 ***
## env.ls$Type                        0.001 ***
## env.ls$Diet:env.ls$Density         0.243
## env.ls$Diet:env.ls$Timepoint       0.001 ***
## env.ls$Density:env.ls$Timepoint    0.001 ***
## env.ls$Diet:env.ls$Type            0.001 ***
## env.ls$Density:env.ls$Type         0.013 *
## env.ls$Timepoint:env.ls$Type       0.001 ***
## env.ls$Diet:env.ls$Density:env.ls$Timepoint 0.001 ***
## env.ls$Diet:env.ls$Density:env.ls$Type    0.051 .
## env.ls$Diet:env.ls$Timepoint:env.ls$Type  0.001 ***
## env.ls$Density:env.ls$Timepoint:env.ls$Type 0.009 **
## env.ls$Diet:env.ls$Density:env.ls$Timepoint:env.ls$Type 0.060 .
## Residual
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

# goodness of fit
stressplot(dbrDA.ls) # good fit.

```



```
# dispersion test
anova(betadisper(wuf.ls, group = interaction(env.ls$Diet, env.ls$Density,
      env.ls$Timepoint, env.ls$Type, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.ls$ContainerID,
          type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.ls, group = interaction(env.ls$Diet, env.ls$Density, :
## some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: Distances
```

```
##           Df Sum Sq   Mean Sq F value    Pr(>F)
## Groups      53 1.1796 0.0222560   2.4057 1.317e-05 ***
## Residuals  162 1.4987 0.0092512
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

3.2. Chicken feed

```

# Calculate dbRDA
dbRDA.cf.ls <- dbrda(wuf.cf ~ env.cf$Density * env.cf$Timepoint * env.cf$Type)
RsquareAdj(dbRDA.cf.ls)

## $r.squared
## [1] 0.5837389
##
## $adj.r.squared
## [1] 0.4526938

# permutation tests
dbRDA.cf.ls0 <- dbrda(wuf.cf ~ 1)
anova(dbRDA.cf.ls0, dbRDA.cf.ls,
       permutations = how(within = Within(type = "none"),
                          plots = Plots(strata = env.cf$ContainerID,
                                         type = "free"), nperm = 999))

## Permutation tests for dbrda under reduced model
## Plots: env.cf$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.cf ~ 1
## Model 2: wuf.cf ~ env.cf$Density * env.cf$Timepoint * env.cf$Type
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     71           7.8921
## 2     54           3.2852 17     4.6069 4.4545 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

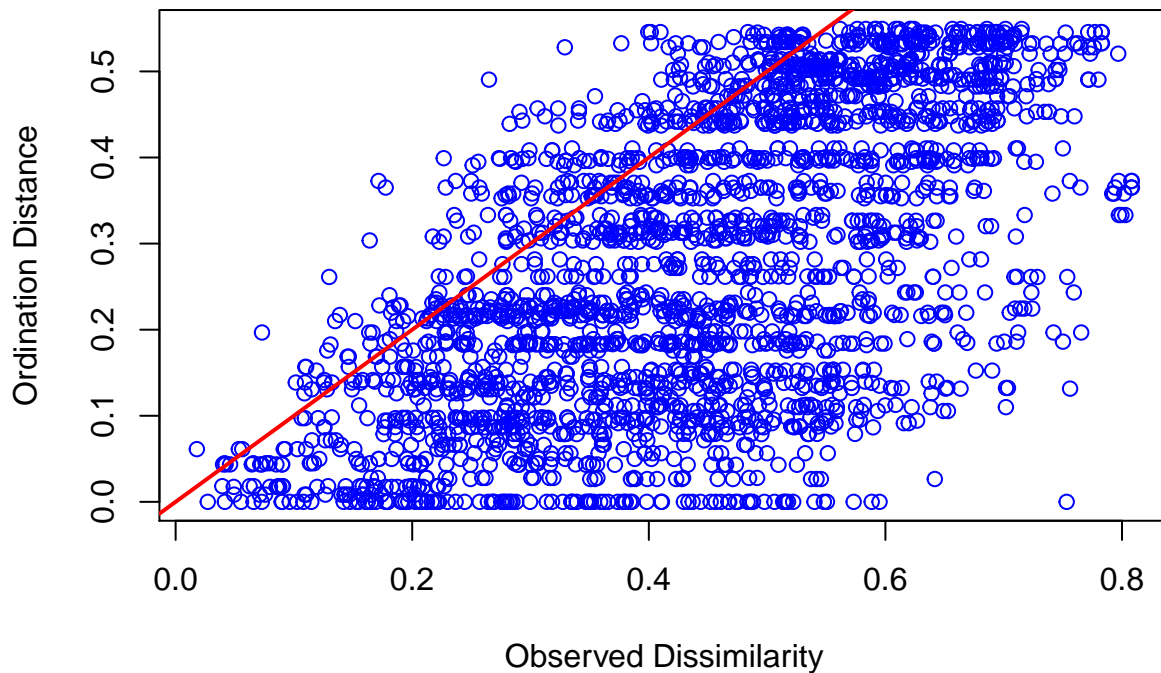
# by terms
aov.cf.ls <- anova.cca(dbRDA.cf.ls, model = "direct", by="terms",
                      permutations = how(within = Within(type = "none"),
                                         plots = Plots(strata = env.cf$ContainerID,
                                                         type = "free"), nperm = 999))
aov.cf.ls

## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cf$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.cf ~ env.cf$Density * env.cf$Timepoint * env.cf$Type)
##
##               Df SumOfSqs      F Pr(>F)
## env.cf$Density  2   0.5259  4.3218 0.001 ***
## env.cf$Timepoint 2   2.5021 20.5643 0.001 ***
## env.cf$Type      1   0.2117  3.4790 0.008 **
## env.cf$Density:env.cf$Timepoint 4   1.1124  4.5712 0.001 ***
## env.cf$Density:env.cf$Type      2   0.0332  0.2729 0.967
## env.cf$Timepoint:env.cf$Type     2   0.1245  1.0235 0.283

```

```
## env.cf$Density:env.cf$Timepoint:env.cf$Type 4 0.0972 0.3993 0.968
## Residual 54 3.2852
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cf.ls) # fit OK
```



```
# dispersion test
anova(betadisper(wuf.cf, group = interaction(env.cf$Density,
env.cf$Timepoint, env.cf$Type, drop = T)),
permutations = how(within = Within(type = "none"),
plots = Plots(strata = env.cf$ContainerID,
type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.cf, group = interaction(env.cf$Density,
## env.cf$Timepoint, : some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
##
## Response: Distances
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups 17 0.42749 0.0251462 3.6159 0.000159 ***
## Residuals 54 0.37554 0.0069544
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

3.3. Camelina

```
# Calculate dbRDA
```

```
dbRDA.cs.ls <- dbrda(wuf.cs ~ env.cs$Density * env.cs$Timepoint * env.cs$Type)  
RsquareAdj(dbRDA.cs.ls)
```

```
## $r.squared  
## [1] 0.5196701  
##  
## $adj.r.squared  
## [1] 0.3684551
```

```
# permutation tests
```

```
dbRDA.cs.ls0 <- dbrda(wuf.cs ~ 1)  
anova(dbRDA.cs.ls0, dbRDA.cs.ls,  
       permutations = how(within = Within(type = "none"),  
                           plots = Plots(strata = env.cs$ContainerID,  
                                           type = "free"), nperm = 999))
```

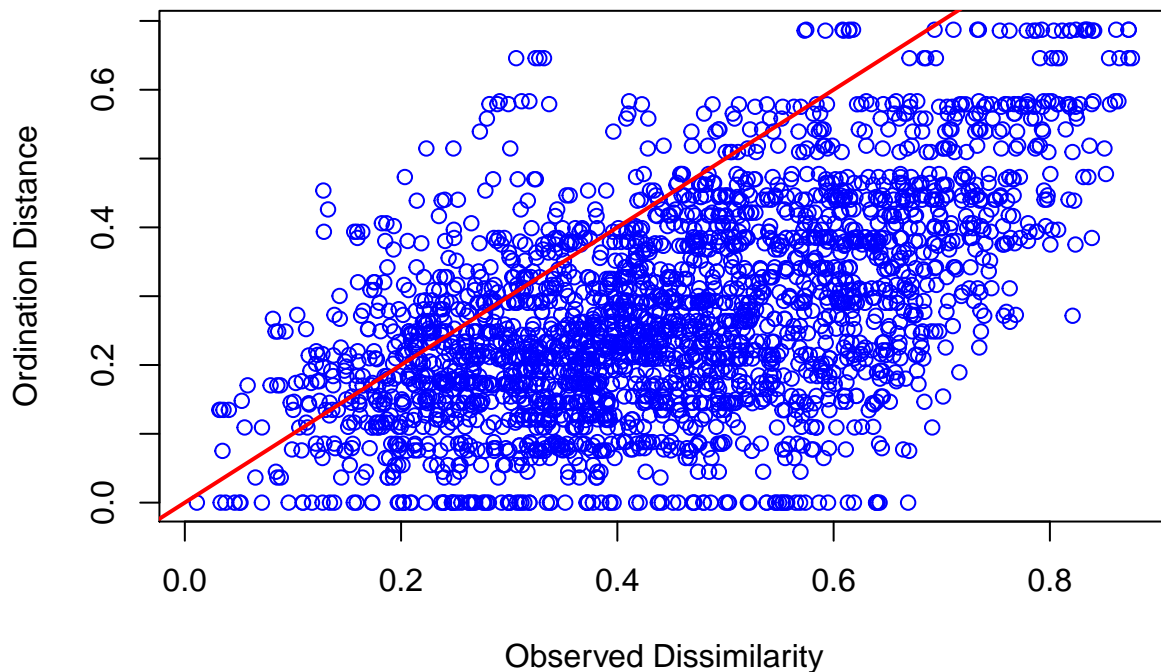
```
## Permutation tests for dbrda under reduced model  
## Plots: env.cs$ContainerID, plot permutation: free  
## Permutation: none  
## Number of permutations: 999  
##  
## Model 1: wuf.cs ~ 1  
## Model 2: wuf.cs ~ env.cs$Density * env.cs$Timepoint * env.cs$Type  
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)  
## 1     71          8.2567  
## 2     54          3.9660 17    4.2908 3.4366 0.001 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.cs.ls <- anova.cca(dbRDA.cs.ls, model = "direct", by="terms",  
                      permutations = how(within = Within(type = "none"),  
                                          plots = Plots(strata = env.cs$ContainerID,  
                                                          type = "free"), nperm = 999))  
aov.cs.ls
```

```
## Permutation test for dbrda under direct model  
## Terms added sequentially (first to last)  
## Plots: env.cs$ContainerID, plot permutation: free  
## Permutation: none  
## Number of permutations: 999  
##  
## Model: dbrda(formula = wuf.cs ~ env.cs$Density * env.cs$Timepoint * env.cs$Type)  
##  
##               Df SumOfSqs      F Pr(>F)  
## env.cs$Density  2  0.5442  3.7046 0.006 **  
## env.cs$Timepoint 2  1.5125 10.2971 0.001 ***
```

```
## env.cs$Type 1 0.4786 6.5166 0.005 **
## env.cs$Density:env.cs$Timepoint 4 0.7378 2.5114 0.004 **
## env.cs$Density:env.cs$Type 2 0.1226 0.8349 0.467
## env.cs$Timepoint:env.cs$Type 2 0.5426 3.6937 0.001 ***
## env.cs$Density:env.cs$Timepoint:env.cs$Type 4 0.3525 1.2000 0.182
## Residual 54 3.9660
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cs.ls) # fit OK.
```



```
# dispersion test
anova(betadisper(wuf.cs, group = interaction(env.cs$Density,
env.cs$Timepoint, env.cs$Type, drop = T)),
permutations = how(within = Within(type = "none"),
plots = Plots(strata = env.cs$ContainerID,
type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.cs, group = interaction(env.cs$Density,
## env.cs$Timepoint, : some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
##
## Response: Distances
```



```
##           Df Sum Sq Mean Sq F value Pr(>F)
## Groups    17 0.34926 0.020545  1.5078 0.1273
## Residuals  54 0.73578 0.013626
```

```
# no significant differences in dispersion.
```

3.4. Chicken manure

```
# Calculate dbRDA
dbRDA.cm.ls <- dbrda(wuf.cm ~ env.cm$Density * env.cm$Timepoint * env.cm$Type)
RsquareAdj(dbRDA.cm.ls)
```

```
## $r.squared
## [1] 0.7326144
##
## $adj.r.squared
## [1] 0.6484375
```

```
# permutation tests
dbRDA.cm.ls0 <- dbrda(wuf.cm ~ 1)
anova(dbRDA.cm.ls0, dbRDA.cm.ls,
      permutations = how(within = Within(type = "none"),
                        plots = Plots(strata = env.cm$ContainerID,
                                     type = "free"), nperm = 999))
```

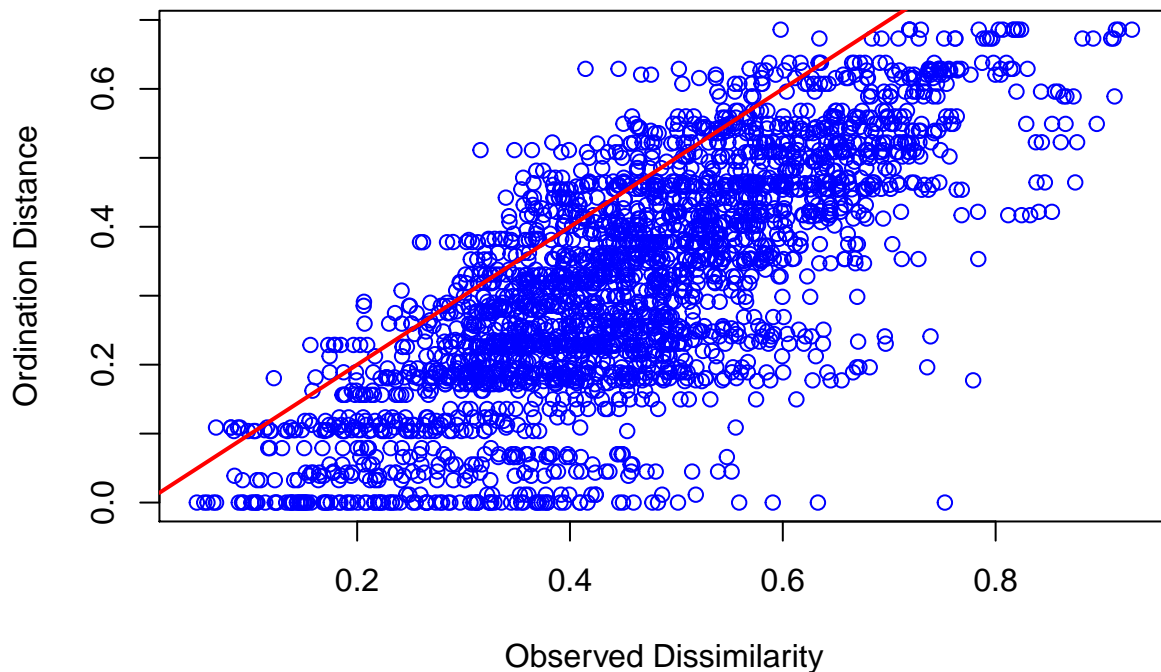
```
## Permutation tests for dbrda under reduced model
## Plots: env.cm$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.cm ~ 1
## Model 2: wuf.cm ~ env.cm$Density * env.cm$Timepoint * env.cm$Type
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     71          8.3204
## 2     54          2.2248 17    6.0957 8.7033 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.cm.ls <- anova.cca(dbRDA.cm.ls, model = "direct", by="terms",
                      permutations = how(within = Within(type = "none"),
                                        plots = Plots(strata = env.cm$ContainerID,
                                                       type = "free"), nperm = 999))
aov.cm.ls
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cm$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
```

```
## Model: dbrda(formula = wuf.cm ~ env.cm$Density * env.cm$Timepoint * env.cm$Type)
##
##          Df SumOfSqs      F Pr(>F)
## env.cm$Density      2  0.28508   3.4598 0.001 ***
## env.cm$Timepoint     2  3.04249  36.9240 0.001 ***
## env.cm$Type          1  1.17820  28.5976 0.001 ***
## env.cm$Density:env.cm$Timepoint    4  0.45705   2.7734 0.009 **
## env.cm$Density:env.cm$Type         2  0.28261   3.4299 0.003 **
## env.cm$Timepoint:env.cm$Type        2  0.61620   7.4783 0.001 ***
## env.cm$Density:env.cm$Timepoint:env.cm$Type  4  0.23403   1.4201 0.148
## Residual              54  2.22476
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cm.ls) # fit good.
```



```
# dispersion test
anova(betadisper(wuf.cm, group = interaction(env.cm$Density,
      env.cm$Timepoint, env.cm$Type, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.cm$ContainerID,
          type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.cm, group = interaction(env.cm$Density,
## env.cm$Timepoint, : some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
##
## Response: Distances
##           Df Sum Sq Mean Sq F value Pr(>F)
## Groups    17 0.28280 0.016635  2.2025 0.01455 *
## Residuals  54 0.40786 0.007553
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

4. dbRDA: larvae

Supplementary Tables S5 and S6 in the manuscript submitted to *Applied and Environmental Microbiology*.
Not included in manuscript Chapter 3 in PhD thesis.

4.1. Total

```
# calculate dbRDA
dbRDA.wuf.1 <- dbrda(wuf.1 ~ env.1$Diet * env.1$Density * env.1$Timepoint)

dbRDA.wuf.1

## Call: dbrda(formula = wuf.1 ~ env.1$Diet * env.1$Density *
## env.1$Timepoint)
##
##              Inertia Proportion Rank RealDims
## Total          18.4663      1.0000
## Constrained    13.0353      0.7059    26      18
## Unconstrained   5.4309      0.2941    81      42
## Inertia is squared Unknown distance
##
## Eigenvalues for constrained axes:
## dbRDA1 dbRDA2 dbRDA3 dbRDA4 dbRDA5 dbRDA6 dbRDA7 dbRDA8 dbRDA9 dbRDA10
##  5.864  2.735  1.289  0.970  0.778  0.432  0.260  0.237  0.170  0.138
## dbRDA11 dbRDA12 dbRDA13 dbRDA14 dbRDA15 dbRDA16 dbRDA17 dbRDA18 idbRDA1 idbRDA2
##  0.091  0.082  0.056  0.034  0.029  0.016  0.012  0.011 -0.002 -0.007
## idbRDA3 idbRDA4 idbRDA5 idbRDA6 idbRDA7 idbRDA8
## -0.010 -0.016 -0.019 -0.021 -0.034 -0.060
##
## Eigenvalues for unconstrained axes:
## MDS1  MDS2  MDS3  MDS4  MDS5  MDS6  MDS7  MDS8
## 1.6724 0.7337 0.6600 0.6027 0.4887 0.3693 0.2658 0.2133
## (Showing 8 of 81 unconstrained eigenvalues)
```

```
RsquareAdj(dbRDA.wuf.1)
```

```
## $r.squared
## [1] 0.7058987
```

```
##
## $adj.r.squared
## [1] 0.6114958
```

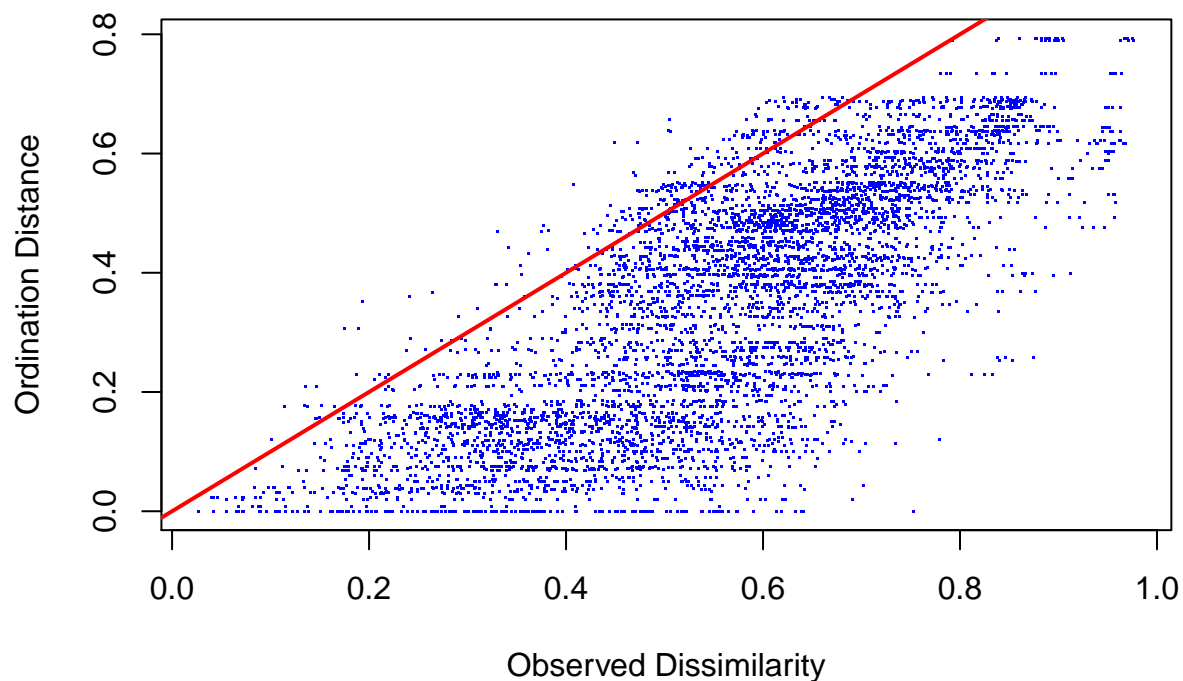
```
# permutation tests
dbRDA.wuf0.1 <- dbrda(wuf.1 ~ 1)
anova.cca(dbRDA.wuf0.1, dbRDA.wuf.1,
           permutations = how(within = Within(type = "none"),
                              plots = Plots(strata = env.1$ContainerID,
                                             type = "free"), nperm = 999))
```

```
## Permutation tests for dbrda under reduced model
## Plots: env.1$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.1 ~ 1
## Model 2: wuf.1 ~ env.1$Diet * env.1$Density * env.1$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1    107      18.4663
## 2     81       5.4309 26   13.035 7.4775 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# by terms
aov.wuf.1 <- anova.cca(dbRDA.wuf.1, model = "direct", by="terms",
                      permutations = how(within = Within(type = "none"),
                                         plots = Plots(strata = env.1$ContainerID,
                                                        type = "free"), nperm = 999))
aov.wuf.1
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.1$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.1 ~ env.1$Diet * env.1$Density * env.1$Timepoint)
##
##           Df SumOfSqs      F Pr(>F)
## env.1$Diet    2   6.4004 47.7291 0.001 ***
## env.1$Density  2   0.1823  1.3598 0.453
## env.1$Timepoint 2   1.2408  9.2529 0.001 ***
## env.1$Diet:env.1$Density 4   0.4446  1.6579 0.378
## env.1$Diet:env.1$Timepoint 4   3.4014 12.6826 0.001 ***
## env.1$Density:env.1$Timepoint 4   0.4791  1.7865 0.001 ***
## env.1$Diet:env.1$Density:env.1$Timepoint 8   0.8866  1.6530 0.001 ***
## Residual    81   5.4309
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbRDA.wuf.1) # good fit.
```



```
# dispersion test
anova(betadisper(wuf.l, group = interaction(env.l$Diet,
      env.l$Density, env.l$Timepoint, drop = T)),
      permutations = how(within = Within(type = "none"),
        plots = Plots(strata = env.l$ContainerID,
          type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.l, group = interaction(env.l$Diet, env.l$Density, :
## some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: Distances
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
## Groups    26 0.48404 0.018617  1.7193 0.03446 *
## Residuals 81 0.87706 0.010828
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

4.2. Chicken feed

```
# Calculate dbRDA
dbRDA.cf.wuf.l <- dbrda(wuf.cfl ~ env.cfl$Density * env.cfl$Timepoint)
RsquareAdj(dbRDA.cf.wuf.l)
```

```
## $r.squared
## [1] 0.6050004
##
## $adj.r.squared
## [1] 0.4879634
```

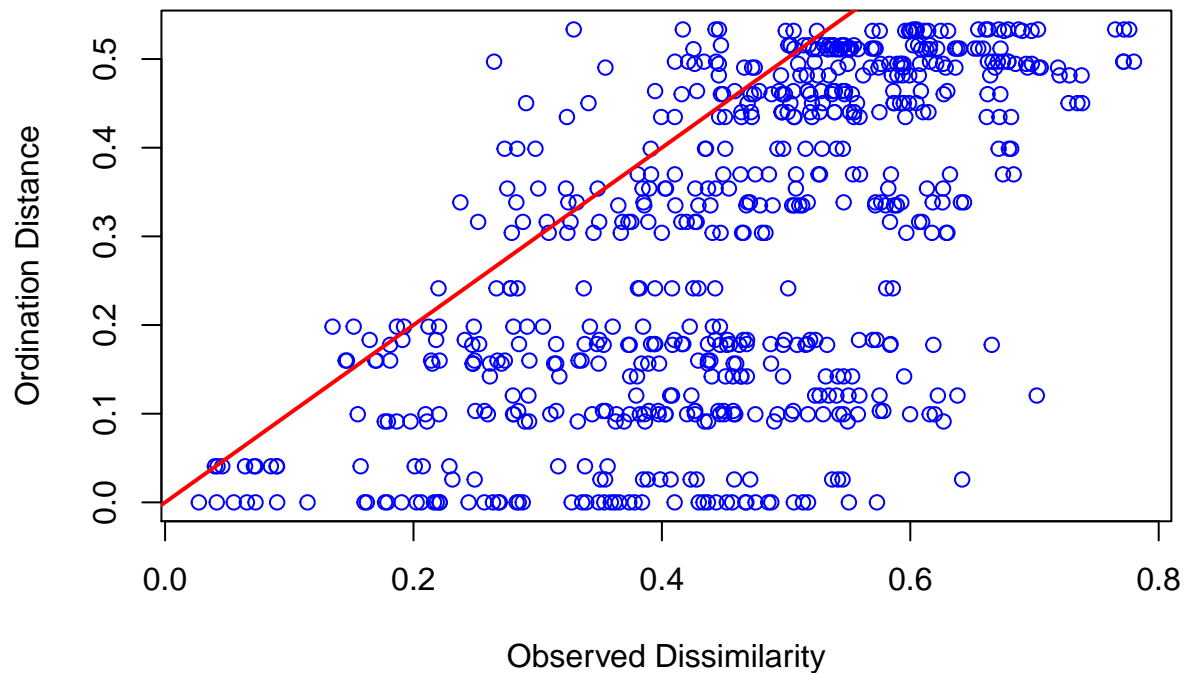
```
# permutation tests
dbRDA.cf.wuf0.l <- dbrda(wuf.cfl ~ 1)
anova(dbRDA.cf.wuf0.l, dbRDA.cf.wuf.l,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.cfl$ContainerID,
                                           type = "free"), nperm = 999))
```

```
## Permutation tests for dbrda under reduced model
## Plots: env.cfl$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.cfl ~ 1
## Model 2: wuf.cfl ~ env.cfl$Density * env.cfl$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     35           4.0209
## 2     27           1.5883  8     2.4327 5.1693 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# by terms
aov.cf.wuf.l <- anova.cca(dbRDA.cf.wuf.l, model = "direct", by="terms",
                          permutations = how(within = Within(type = "none"),
                                              plots = Plots(strata = env.cfl$ContainerID,
                                                            type = "free"), nperm = 999))
aov.cf.wuf.l
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cfl$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.cfl ~ env.cfl$Density * env.cfl$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.cfl$Density  2  0.26069  2.2158 0.001 ***
## env.cfl$Timepoint  2  1.48978 12.6628 0.001 ***
## env.cfl$Density:env.cfl$Timepoint  4  0.68221  2.8993 0.005 **
## Residual          27  1.58827
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cf.wuf.1) # fit OK.
```



```
# dispersion test
anova(betadisper(wuf.cfl, group = interaction(env.cfl$Density,
env.cfl$Timepoint, drop = T)),
permutations = how(within = Within(type = "none"),
plots = Plots(strata = env.cfl$ContainerID,
type = "free"), nperm = 999))
```

```
## Warning in betadisper(wuf.cfl, group = interaction(env.cfl$Density,
## env.cfl$Timepoint, : some squared distances are negative and changed to zero
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: Distances
```

```
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups    8 0.21026 0.0262820  3.5436 0.006281 **
## Residuals 27 0.20025 0.0074167
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# significant differences in dispersion: caution with anova.cca output.
```

4.3. Camelina

```
# Calculate dbRDA
dbRDA.cs.wuf.1 <- dbrda(wuf.csl ~ env.csl$Density * env.csl$Timepoint)
RsquareAdj(dbRDA.cs.wuf.1)
```

```
## $r.squared
## [1] 0.4588386
##
## $adj.r.squared
## [1] 0.2984945
```

```
# permutation tests
dbRDA.cs.wuf0.1 <- dbrda(wuf.csl ~ 1)
anova(dbRDA.cs.wuf0.1, dbRDA.cs.wuf.1,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.csl$ContainerID,
                                           type = "free"), nperm = 999))
```

```
## Permutation tests for dbrda under reduced model
## Plots: env.csl$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.csl ~ 1
## Model 2: wuf.csl ~ env.csl$Density * env.csl$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     35         4.1741      1.9152 2.8616 0.002 **
## 2     27         2.2588      8      1.9152 2.8616 0.002 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

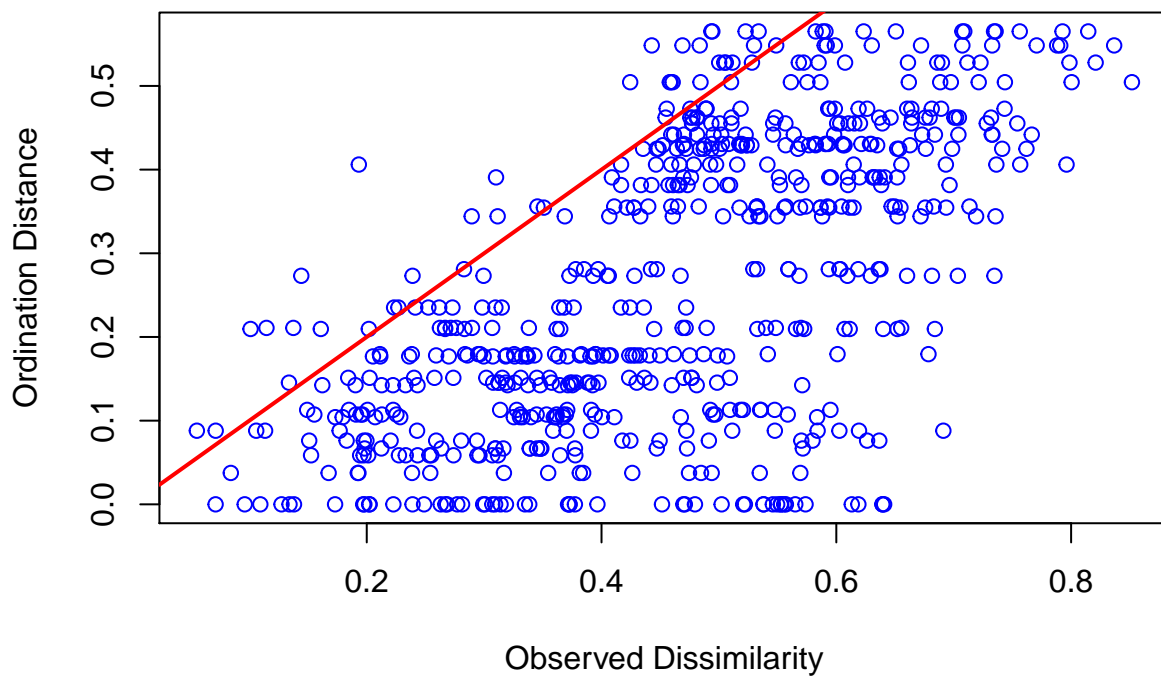
```
# by terms
aov.cs.wuf.1 <- anova.cca(dbRDA.cs.wuf.1, model = "direct", by="terms",
                          permutations = how(within = Within(type = "none"),
                                              plots = Plots(strata = env.csl$ContainerID,
                                                            type = "free"), nperm = 999))
aov.cs.wuf.1
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.csl$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.csl ~ env.csl$Density * env.csl$Timepoint)
##                                     Df SumOfSqs      F Pr(>F)
## env.csl$Density                     2  0.21983 1.3138 0.246
```



```
## env.csl$Timepoint          2  1.47430 8.8112  0.002 **
## env.csl$Density:env.csl$Timepoint  4  0.22109 0.6607  0.707
## Residual                   27  2.25885
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cs.wuf.1) # fit OK.
```



```
# dispersion test
anova(betadisper(wuf.csl, group = interaction(env.csl$Density,
  env.csl$Timepoint, drop = T)),
  permutations = how(within = Within(type = "none"),
    plots = Plots(strata = env.csl$ContainerID,
      type = "free"), nperm = 999))
```

```
## Analysis of Variance Table
##
## Response: Distances
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups      8 0.10291  0.012863   0.7689 0.6327
## Residuals  27 0.45167  0.016729
```

```
# no significant difference in dispersion.
```

4.4. Chicken manure

```
# Calculate dbRDA
dbRDA.cm.wuf.l <- dbrda(wuf.cml ~ env.cml$Density * env.cml$Timepoint)
RsquareAdj(dbRDA.cm.wuf.l)
```

```
## $r.squared
## [1] 0.5908344
##
## $adj.r.squared
## [1] 0.4696002
```

```
# permutation tests
dbRDA.cm.wuf0.l <- dbrda(wuf.cml ~ 1)
anova(dbRDA.cm.wuf0.l, dbRDA.cm.wuf.l,
       permutations = how(within = Within(type = "none"),
                           plots = Plots(strata = env.cml$ContainerID,
                                           type = "free"), nperm = 999))
```

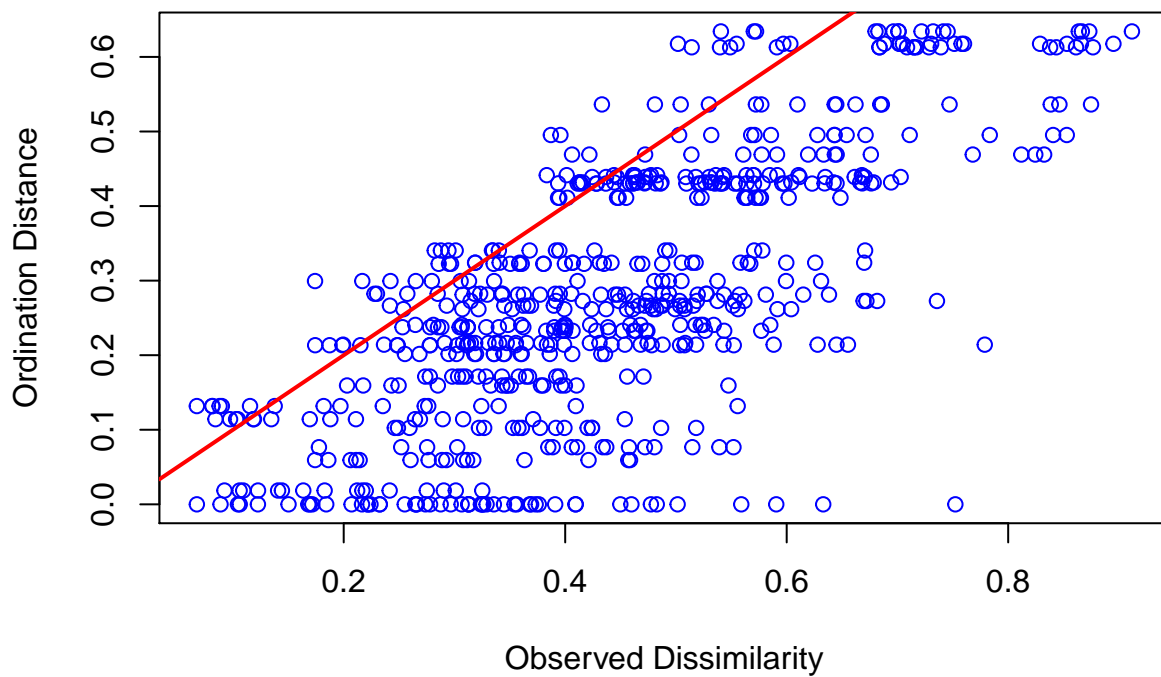
```
## Permutation tests for dbrda under reduced model
## Plots: env.cml$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model 1: wuf.cml ~ 1
## Model 2: wuf.cml ~ env.cml$Density * env.cml$Timepoint
##   ResDf ResChiSquare Df ChiSquare      F Pr(>F)
## 1     35          3.8709
## 2     27          1.5838  8      2.287 4.8735 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# by terms
aov.cm.wuf.l <- anova.cca(dbRDA.cm.wuf.l, model = "direct", by="terms",
                         permutations = how(within = Within(type = "none"),
                                             plots = Plots(strata = env.cml$ContainerID,
                                                             type = "free"), nperm = 999))
aov.cm.wuf.l
```

```
## Permutation test for dbrda under direct model
## Terms added sequentially (first to last)
## Plots: env.cml$ContainerID, plot permutation: free
## Permutation: none
## Number of permutations: 999
##
## Model: dbrda(formula = wuf.cml ~ env.cml$Density * env.cml$Timepoint)
##
##               Df SumOfSqs      F Pr(>F)
## env.cml$Density 2  0.14647  1.2485 0.048 *
```

```
## env.cml$Timepoint          2  1.67810 14.3036  0.001 ***
## env.cml$Density:env.cml$Timepoint  4  0.46247  1.9710  0.059 .
## Residual                   27  1.58383
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# goodness of fit
stressplot(dbrDA.cm.wuf.1) # fit good.
```



```
# dispersion test
anova(betadisper(wuf.cml, group = interaction(env.cml$Density,
  env.cml$Timepoint, drop = T)),
  permutations = how(within = Within(type = "none"),
    plots = Plots(strata = env.cml$ContainerID,
      type = "free"), nperm = 999))
```

```
## Analysis of Variance Table
##
## Response: Distances
##      Df Sum Sq Mean Sq F value Pr(>F)
## Groups      8  0.13025  0.0162808    1.667  0.1526
## Residuals   27  0.26369  0.0097663
```

no significant differences in dispersion.