8. Worksheet: Among Site (Beta) Diversity – Part 2

Jonathan Enriquez Madrid; Z620: Quantitative Biodiversity, Indiana University

08 February, 2023

OVERVIEW

In this worksheet, we continue to explore concepts, statistics, and visualizations related to β-diversity. Now that you know how to formally quantify β-diversity, we will learn how to test hypotheses about β-diversity using multivariate statistics.

Directions:

1. In the Markdown version of this document in your cloned repo, change "Student Name" on line 3 (above) with your name.
2. Complete as much of the worksheet as possible during class.
3. Use the handout as a guide; it contains a more complete description of data sets along with examples of proper scripting needed to carry out the exercises.
4. Answer questions in the worksheet. Space for your answers is provided in this document and is indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio (color may vary if you changed the editor theme).
5. Before you leave the classroom today, it is *imperative* that you **push** this file to your GitHub repo, at whatever stage you are. This will enable you to pull your work onto your own computer.
6. When you have completed the worksheet, **Knit** the text and code into a single PDF file by pressing the `Knit` button in the RStudio scripting panel. This will save the PDF output in your '8.BetaDiversity' folder.
7. After Knitting, please submit the worksheet by making a **push** to your GitHub repo and then create a **pull request** via GitHub. Your pull request should include this file (**8.BetaDiversity_2_Worksheet.Rmd**) with all code blocks filled out and questions answered) and the PDF output of `Knitr` (**8.BetaDiversity_2_Worksheet.pdf**).

The completed exercise is due on **Wednesday, February 8th, 2023 before 12:00 PM (noon)**.

1) R SETUP

Typically, the first thing you will do in either an R script or an RMarkdown file is setup your environment. This includes things such as setting the working directory and loading any packages that you will need.

In the R code chunk below, provide the code to:

1. clear your R environment,
2. print your current working directory,
3. set your working directory to your "*/6.BetaDiversity*" folder, and

4. load the **vegan** R package (be sure to install if needed).

```
install.packages('plyr', repos = "http://cran.us.r-project.org")
```

```
## Installing package into 'C:/Users/jonat/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'plyr' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\jonat\AppData\Local\Temp\RtmpGAuk1w\downloaded_packages
```

```
options(repos = list(CRAN="http://cran.rstudio.com/"))
rm(list = ls())
getwd()
```

```
## [1] "C:/Users/jonat/GitHub/QB2023_Enriquez_Madrid/2.Worksheets/6.BetaDiversity"
```

```
setwd("C:/Users/jonat/GitHub/QB2023_Enriquez_Madrid/2.Worksheets/6.BetaDiversity")
library(vegan)
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```
## This is vegan 2.6-4
```

```
library(ade4)
install.packages("rlang")
```

```
## Installing package into 'C:/Users/jonat/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'rlang' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'rlang'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\jonat\AppData\Local\R\win-library\4.2\00LOCK\rlang\libs\x64\rlang.dll
## to C:\Users\jonat\AppData\Local\R\win-library\4.2\rlang\libs\x64\rlang.dll:
## Permission denied
```

```
## Warning: restored 'rlang'
```

```
##
## The downloaded binary packages are in
##  C:\Users\jonat\AppData\Local\Temp\RtmpGAuk1w\downloaded_packages
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(gplots)
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(BiodiversityR)
```

```
## Loading required package: tcltk
```

```
## BiodiversityR 2.15-1: Use command BiodiversityRGUI() to launch the Graphical User Interface;
## to see changes use BiodiversityRGUI(changeLog=TRUE, backward.compatibility.messages=TRUE)
```

```
library(indicspecies)
```

2) LOADING DATA

Load dataset

In the R code chunk below, load the `doubs` dataset from the `ade4` package

```
# note, pleae do not print the dataset when submitting
data(doubs)
```

3) HYPOTHESIS TESTING

A. Multivariate Procedures for Categorical Designs

Earlier work done in the Doubs River suggested that the river has four distinct regions of habitat quality: the first region (sites 1-14) of "high quality"; the second (sites 15 - 19) and fourth (sites 26 - 30) of "moderate quality"; and the third (sites 20 - 25) of "low quality".

In the code chunk below, test the hypothesis that fish community composition varies with river quality.

1. create a factor vector that categorizes habitat quality in the Doubs River,
2. use the multivariate analyses for categorical predictors to describe how fish community structure relates to habitat quality.

```
fish <- doubs$fish
fish <- fish[-8,]

quality <- c(rep("HQ", 13), rep("MQ", 5), rep("LQ", 6), rep("MQ", 5))
adonis(fish ~ quality, method = "bray", permutations = 999)#quality is sign.
```

```
## 'adonis' will be deprecated: use 'adonis2' instead

## $aov.tab
## Permutation: free
## Number of permutations: 999
## 
## Terms added sequentially (first to last)
## 
##           Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)    
## quality    2    3.0947 1.54733   10.97 0.45765  0.001 ***
## Residuals 26    3.6674 0.14105         0.54235           
## Total     28    6.7621                 1.00000           
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## $call
## adonis(formula = fish ~ quality, permutations = 999, method = "bray")
## 
## $coefficients
##                   Cogo      Satr      Phph      Neba       Thth       Teso
## (Intercept)  0.4384615  1.546154  1.941880  2.191453  0.4230769  0.5948718
## quality1     0.1769231  1.992308  1.750427  1.116239  0.3461538 -0.2102564
## quality2    -0.4384615 -1.546154 -1.608547 -1.524786 -0.4230769 -0.5948718
##                   Chna          Chto       Lele       Lece       Baba
## (Intercept)  0.7777778  1.000000e+00  1.5418803  2.0324786  1.7145299
## quality1    -0.7777778 -1.000000e+00 -0.8495726 -0.8017094 -1.6376068
## quality2     0.5555556 -1.135321e-16 -0.2085470  0.1341880  0.4521368
##                    Spbi       Gogo        Eslu       Pefl Rham       Legi
## (Intercept)  1.03333333  2.1649573  1.46410256  1.3461538  1.3  1.1444444
## quality1    -1.03333333 -1.7034188 -0.77179487 -0.8076923 -1.3 -1.1444444
## quality2    -0.03333333  0.6683761  0.03589744  0.1538462  0.2  0.1888889
##                   Scer        Cyca       Titi       Abbr       Icme       Acce
## (Intercept)  0.8401709  0.96666667  1.7128205  1.0444444  0.6666667  1.5333333
## quality1    -0.6863248 -0.96666667 -1.1743590 -1.0444444 -0.6666667 -1.5333333
## quality2     0.3264957  0.03333333  0.2871795  0.2888889 -0.1666667  0.4666667
##                   Ruru       Blbj      Alal        Anan
## (Intercept)  2.4452991  1.2555556  2.455556  1.03333333
## quality1    -1.6760684 -1.2555556 -2.455556 -1.03333333
## quality2     0.7213675  0.4111111  1.711111 -0.03333333
## 
## $coef.sites
##                      1          2          3          4           5          6
## (Intercept)  0.8837625  0.7158590  0.6938684  0.6230418 0.593432800  0.5790159
## quality1    -0.1866775 -0.3635334 -0.3456333 -0.2543114 0.001651813 -0.2025678
## quality2     0.1162375  0.2564578  0.2548711  0.2057479 0.056023013  0.1732300
##                      7           9         10         11         12         13
## (Intercept)  0.6581737  0.71175827  0.6442621  0.7094992  0.6760632  0.7319489
## quality1    -0.3307310 -0.05986164 -0.2179352 -0.2948057 -0.3340042 -0.3052481
## quality2     0.2381097  0.04794668  0.1706838  0.2036504  0.2492646  0.2428243
##                     14         15          16         17        18         19
## (Intercept)  0.6625054  0.6126096  0.58629885 0.53831620 0.5265411  0.5207733
## quality1    -0.2452690 -0.1465108 -0.01793157 0.06448519 0.1353961  0.1853921
## quality2     0.2192664  0.2018642  0.14094735 0.08434935 0.0300669 -0.0183066
##                     20         21         22         23         24         25
## (Intercept)  0.5199318  0.5323960  0.5576307  0.8035439  0.6877772  0.6964587
## quality1     0.2948405  0.3155710  0.3206040  0.1145069  0.2110503  0.1691525
## quality2    -0.1191038 -0.1333993 -0.1281596 -0.1892430 -0.2071918 -0.1908893
##                      26         27         28          29          30
## (Intercept)  0.5546865  0.5573809  0.5683021  0.56565435  0.61089065
## quality1     0.2955016  0.3181323  0.3117246  0.28417790  0.30341313
## quality2    -0.1264475 -0.1150663 -0.1044695 -0.07583091 -0.09189382
##
## $f.perms
##              [,1]
##    [1,] 0.3511434
##    [2,] 1.0222747
##    [3,] 0.8352058
##    [4,] 0.7519230
##    [5,] 0.6798512
##    [6,] 1.8889769
##    [7,] 0.9329910
##    [8,] 1.6639570
##    [9,] 0.8974968
##   [10,] 0.6031947
##   [11,] 0.5812851
##   [12,] 1.2842396
##   [13,] 0.9796231
##   [14,] 0.8074513
##   [15,] 2.2658006
##   [16,] 1.0967800
##   [17,] 1.2939774
##   [18,] 0.9635370
##   [19,] 2.1784030
##   [20,] 0.7806093
##   [21,] 0.8280378
##   [22,] 0.5474028
##   [23,] 0.7173088
##   [24,] 0.6571276
##   [25,] 4.6677653
##   [26,] 0.6088349
##   [27,] 0.2884388
##   [28,] 0.3899238
##   [29,] 0.7712878
##   [30,] 1.4306569
##   [31,] 1.3219456
##   [32,] 0.7053304
##   [33,] 0.8422472
##   [34,] 0.5950623
##   [35,] 2.0350944
##   [36,] 0.5123535
##   [37,] 1.0752037
##   [38,] 1.2646965
##   [39,] 0.9094961
##   [40,] 2.1825782
##   [41,] 0.9906747
##   [42,] 0.5232698
##   [43,] 1.2245877
##   [44,] 1.6264834
##  [45,] 0.5569538
##  [46,] 0.5491310
##  [47,] 1.7791309
##  [48,] 0.3007732
##  [49,] 0.9650478
##  [50,] 1.1112267
##  [51,] 1.2379623
##  [52,] 0.4047540
##  [53,] 0.9154487
##  [54,] 0.5056026
##  [55,] 0.4558493
##  [56,] 2.7131992
##  [57,] 1.6318780
##  [58,] 0.6568183
##  [59,] 1.3908992
##  [60,] 1.1551814
##  [61,] 1.0136279
##  [62,] 0.4759840
##  [63,] 0.9338644
##  [64,] 0.6805657
##  [65,] 1.7196533
##  [66,] 0.9840083
##  [67,] 1.3767961
##  [68,] 0.8039434
##  [69,] 0.7065195
##  [70,] 0.5796043
##  [71,] 1.5642729
##  [72,] 0.3694479
##  [73,] 0.5076214
##  [74,] 1.0797221
##  [75,] 0.6031213
##  [76,] 1.6597043
##  [77,] 0.5802660
##  [78,] 0.8618882
##  [79,] 1.6786655
##  [80,] 0.6734182
##  [81,] 0.4583991
##  [82,] 0.7259252
##  [83,] 2.2979806
##  [84,] 0.9364903
##  [85,] 1.4221961
##  [86,] 1.4259617
##  [87,] 0.5233548
##  [88,] 0.7001832
##  [89,] 0.6496090
##  [90,] 0.6692543
##  [91,] 0.3311008
##  [92,] 0.8991814
##  [93,] 0.9994271
##  [94,] 0.5593194
##  [95,] 0.5806798
##  [96,] 0.5794375
##  [97,] 1.0154862
##  [98,] 0.9988540
##   [99,] 0.7394153
##  [100,] 1.2292926
##  [101,] 1.0992833
##  [102,] 1.3576165
##  [103,] 1.3197640
##  [104,] 1.0261583
##  [105,] 0.8644525
##  [106,] 2.7247814
##  [107,] 0.8111697
##  [108,] 0.8693144
##  [109,] 0.8594757
##  [110,] 1.3317411
##  [111,] 0.2198500
##  [112,] 0.3147381
##  [113,] 2.2594803
##  [114,] 0.3532077
##  [115,] 2.3067428
##  [116,] 2.0212813
##  [117,] 0.6761011
##  [118,] 0.8366472
##  [119,] 0.7631631
##  [120,] 1.4866762
##  [121,] 0.3695215
##  [122,] 0.4032999
##  [123,] 0.4228585
##  [124,] 0.8851498
##  [125,] 0.6610990
##  [126,] 0.7643412
##  [127,] 1.0506666
##  [128,] 0.7315430
##  [129,] 0.6219223
##  [130,] 0.8992961
##  [131,] 0.9198830
##  [132,] 2.0633214
##  [133,] 1.6563948
##  [134,] 1.0289473
##  [135,] 0.6876315
##  [136,] 1.7223921
##  [137,] 0.8810459
##  [138,] 1.2414866
##  [139,] 0.8784810
##  [140,] 0.7892791
##  [141,] 1.0850923
##  [142,] 1.8710496
##  [143,] 1.2128880
##  [144,] 0.4771572
##  [145,] 0.8229440
##  [146,] 0.4406908
##  [147,] 2.0620812
##  [148,] 0.9189313
##  [149,] 1.5121096
##  [150,] 0.9909195
##  [151,] 1.1033278
##  [152,] 1.0231679
##  [153,] 0.8430012
##  [154,] 0.9791751
##  [155,] 0.5958162
##  [156,] 0.7344313
##  [157,] 0.7268576
##  [158,] 0.5517567
##  [159,] 1.0771945
##  [160,] 0.6479774
##  [161,] 0.5003068
##  [162,] 0.3809114
##  [163,] 0.8304503
##  [164,] 0.7164501
##  [165,] 1.0297369
##  [166,] 0.6018847
##  [167,] 4.0316709
##  [168,] 0.9803144
##  [169,] 2.3837895
##  [170,] 2.8912604
##  [171,] 1.4989581
##  [172,] 1.0660509
##  [173,] 1.0249142
##  [174,] 1.3523108
##  [175,] 1.5287602
##  [176,] 1.6483047
##  [177,] 1.6515492
##  [178,] 1.4552490
##  [179,] 0.9450910
##  [180,] 3.0584556
##  [181,] 0.6171640
##  [182,] 0.3574491
##  [183,] 1.2356741
##  [184,] 0.5813832
##  [185,] 0.3674797
##  [186,] 1.4533319
##  [187,] 1.6784291
##  [188,] 1.2262863
##  [189,] 0.9035176
##  [190,] 0.5265902
##  [191,] 0.7722654
##  [192,] 0.3848942
##  [193,] 0.5766927
##  [194,] 1.1258574
##  [195,] 2.2873341
##  [196,] 0.8832842
##  [197,] 1.7423239
##  [198,] 0.9278701
##  [199,] 0.5523261
##  [200,] 0.3732422
##  [201,] 0.4440406
##  [202,] 1.2146872
##  [203,] 0.2312302
##  [204,] 0.8252763
##  [205,] 1.0016199
##  [206,] 2.6717601
## [207,] 1.1182959
## [208,] 0.3228391
## [209,] 1.4724076
## [210,] 0.5896614
## [211,] 0.8741290
## [212,] 1.1285371
## [213,] 1.7783814
## [214,] 0.9278860
## [215,] 0.8005685
## [216,] 0.9933652
## [217,] 0.9136197
## [218,] 0.3743704
## [219,] 0.5123023
## [220,] 0.6974704
## [221,] 0.5953753
## [222,] 0.6021610
## [223,] 0.4895964
## [224,] 1.0022304
## [225,] 1.3094589
## [226,] 0.7509760
## [227,] 1.2778784
## [228,] 0.7703410
## [229,] 2.6816019
## [230,] 1.1614989
## [231,] 1.6884831
## [232,] 1.4951561
## [233,] 0.4688212
## [234,] 0.5374195
## [235,] 0.6341669
## [236,] 0.9665911
## [237,] 0.5530757
## [238,] 0.6832468
## [239,] 0.8491904
## [240,] 0.8465342
## [241,] 0.8381826
## [242,] 0.3798952
## [243,] 1.2339437
## [244,] 0.7528727
## [245,] 0.8764512
## [246,] 1.1476483
## [247,] 2.1669179
## [248,] 1.4868472
## [249,] 0.7836719
## [250,] 0.9693001
## [251,] 0.5525703
## [252,] 0.5622556
## [253,] 1.1430095
## [254,] 1.5543332
## [255,] 0.7805078
## [256,] 0.1955238
## [257,] 0.5327526
## [258,] 1.9293255
## [259,] 1.0864532
## [260,] 0.8452598
##  [261,] 1.7237570
##  [262,] 0.5447413
##  [263,] 0.7584041
##  [264,] 1.2243886
##  [265,] 1.2849653
##  [266,] 1.2941241
##  [267,] 0.9770350
##  [268,] 1.5434362
##  [269,] 0.9629685
##  [270,] 0.6944529
##  [271,] 0.4272729
##  [272,] 1.1056264
##  [273,] 0.6064211
##  [274,] 1.9969207
##  [275,] 0.4565969
##  [276,] 1.6244215
##  [277,] 1.6829002
##  [278,] 1.3536691
##  [279,] 0.7801863
##  [280,] 1.6000794
##  [281,] 0.7036226
##  [282,] 1.1205794
##  [283,] 1.2771368
##  [284,] 1.1604334
##  [285,] 0.8186231
##  [286,] 3.1243171
##  [287,] 0.5395528
##  [288,] 0.4735861
##  [289,] 0.2953071
##  [290,] 1.0460589
##  [291,] 0.6869453
##  [292,] 0.6420916
##  [293,] 0.3868376
##  [294,] 0.7546171
##  [295,] 0.5840119
##  [296,] 0.7150695
##  [297,] 1.5101195
##  [298,] 1.1328904
##  [299,] 0.5226233
##  [300,] 1.0365059
##  [301,] 0.6078103
##  [302,] 1.1149333
##  [303,] 1.0496701
##  [304,] 0.8874169
##  [305,] 0.7988642
##  [306,] 1.3091663
##  [307,] 1.4389365
##  [308,] 2.9459773
##  [309,] 1.0683488
##  [310,] 0.2835461
##  [311,] 1.7625835
##  [312,] 0.8109952
##  [313,] 0.8261467
##  [314,] 0.7628452
##  [315,] 0.3372115
##  [316,] 0.3358894
##  [317,] 3.4447224
##  [318,] 0.7177553
##  [319,] 3.1479631
##  [320,] 0.8047090
##  [321,] 0.4140076
##  [322,] 0.9168031
##  [323,] 0.5708131
##  [324,] 1.0062177
##  [325,] 0.8183769
##  [326,] 0.3759031
##  [327,] 1.6275899
##  [328,] 1.7800337
##  [329,] 0.4543523
##  [330,] 1.8910496
##  [331,] 0.4605310
##  [332,] 0.9264341
##  [333,] 1.7531792
##  [334,] 0.2183221
##  [335,] 0.8026409
##  [336,] 1.4560368
##  [337,] 2.0142615
##  [338,] 0.5649226
##  [339,] 3.0588666
##  [340,] 1.7334099
##  [341,] 1.3090952
##  [342,] 0.5506423
##  [343,] 1.3867927
##  [344,] 0.8712723
##  [345,] 0.6522046
##  [346,] 0.4780178
##  [347,] 1.4745341
##  [348,] 0.8778942
##  [349,] 0.8709148
##  [350,] 1.1637402
##  [351,] 0.5408109
##  [352,] 1.2816671
##  [353,] 0.7960125
##  [354,] 0.9234318
##  [355,] 1.3708970
##  [356,] 2.2137229
##  [357,] 1.7203789
##  [358,] 0.8450196
##  [359,] 0.7856235
##  [360,] 0.2719594
##  [361,] 0.4488170
##  [362,] 0.7984078
##  [363,] 1.8603679
##  [364,] 0.3639876
##  [365,] 0.5627305
##  [366,] 0.6745080
##  [367,] 1.8573549
##  [368,] 0.6429008
## [369,] 0.5487785
## [370,] 0.6304503
## [371,] 0.8967277
## [372,] 0.3693241
## [373,] 1.2651971
## [374,] 0.7236476
## [375,] 0.8830375
## [376,] 0.9653961
## [377,] 2.0497729
## [378,] 0.6409717
## [379,] 1.0702954
## [380,] 0.9027270
## [381,] 0.3028390
## [382,] 1.2264469
## [383,] 1.4657780
## [384,] 0.8110616
## [385,] 0.4585579
## [386,] 1.1828732
## [387,] 0.9490148
## [388,] 0.7789067
## [389,] 0.5528744
## [390,] 0.6047803
## [391,] 0.3902495
## [392,] 2.7847150
## [393,] 0.6848809
## [394,] 1.2400716
## [395,] 1.0370737
## [396,] 0.7812239
## [397,] 1.2182017
## [398,] 1.8084353
## [399,] 0.7149755
## [400,] 0.2962071
## [401,] 1.8912961
## [402,] 0.7786116
## [403,] 0.8408199
## [404,] 1.0199794
## [405,] 0.9312166
## [406,] 0.9428947
## [407,] 0.7873108
## [408,] 1.3240131
## [409,] 0.7062754
## [410,] 1.6769910
## [411,] 1.8974345
## [412,] 0.5132315
## [413,] 0.6700029
## [414,] 1.2135464
## [415,] 0.6209677
## [416,] 0.3210936
## [417,] 0.7913323
## [418,] 1.0982132
## [419,] 0.5840557
## [420,] 0.8450877
## [421,] 0.4043131
## [422,] 0.5339514
##  [423,] 1.0230869
##  [424,] 1.2310667
##  [425,] 1.7433005
##  [426,] 2.3365038
##  [427,] 1.3316246
##  [428,] 0.9469773
##  [429,] 0.9298055
##  [430,] 0.9353411
##  [431,] 2.8705323
##  [432,] 0.9585368
##  [433,] 3.0033602
##  [434,] 1.4355834
##  [435,] 2.1453810
##  [436,] 0.8942634
##  [437,] 2.5344080
##  [438,] 0.7885647
##  [439,] 1.1548896
##  [440,] 1.5388091
##  [441,] 0.7883193
##  [442,] 1.2694890
##  [443,] 0.8029171
##  [444,] 0.8775228
##  [445,] 0.6489454
##  [446,] 0.8062995
##  [447,] 1.4778244
##  [448,] 1.9818572
##  [449,] 0.2204154
##  [450,] 0.8060059
##  [451,] 0.8174823
##  [452,] 0.8201195
##  [453,] 2.6431014
##  [454,] 2.7083591
##  [455,] 1.7394759
##  [456,] 3.0188493
##  [457,] 0.7676979
##  [458,] 0.9336820
##  [459,] 0.8681450
##  [460,] 0.8794012
##  [461,] 0.9167194
##  [462,] 0.9205459
##  [463,] 1.0139373
##  [464,] 0.5990975
##  [465,] 0.9848916
##  [466,] 0.3287591
##  [467,] 0.4773433
##  [468,] 1.6541414
##  [469,] 1.0595712
##  [470,] 2.1568477
##  [471,] 0.5505023
##  [472,] 0.6651495
##  [473,] 1.4761233
##  [474,] 0.4332774
##  [475,] 0.8172905
##  [476,] 0.9648459
##  [477,] 3.4829745
##  [478,] 1.1423838
##  [479,] 0.7373681
##  [480,] 0.5094479
##  [481,] 0.6253535
##  [482,] 1.9081924
##  [483,] 0.9971366
##  [484,] 1.3690782
##  [485,] 0.3771120
##  [486,] 0.7641634
##  [487,] 0.8872098
##  [488,] 0.9313145
##  [489,] 0.3470228
##  [490,] 1.1584790
##  [491,] 1.4359356
##  [492,] 0.3426792
##  [493,] 1.0752513
##  [494,] 1.3592774
##  [495,] 0.6126462
##  [496,] 1.5509487
##  [497,] 0.6194198
##  [498,] 0.3832431
##  [499,] 1.3807682
##  [500,] 0.5907169
##  [501,] 0.5714345
##  [502,] 1.4095072
##  [503,] 1.1540272
##  [504,] 0.6236525
##  [505,] 2.3609713
##  [506,] 1.2076638
##  [507,] 0.7219004
##  [508,] 3.9741605
##  [509,] 1.2507348
##  [510,] 0.3167721
##  [511,] 0.9118206
##  [512,] 0.6964129
##  [513,] 1.4416232
##  [514,] 1.0770823
##  [515,] 1.3887511
##  [516,] 0.5327117
##  [517,] 0.4366706
##  [518,] 0.7705354
##  [519,] 0.3400056
##  [520,] 0.6951536
##  [521,] 0.8189818
##  [522,] 0.7999054
##  [523,] 1.7415996
##  [524,] 2.7893359
##  [525,] 0.6955859
##  [526,] 0.7396591
##  [527,] 0.3911904
##  [528,] 1.3244591
##  [529,] 3.3319032
##  [530,] 0.4229165
##  [531,] 1.7093854
##  [532,] 2.0197838
##  [533,] 1.9589302
##  [534,] 0.7397024
##  [535,] 0.5771591
##  [536,] 1.4279013
##  [537,] 1.2807859
##  [538,] 1.5231112
##  [539,] 1.2351355
##  [540,] 0.6283394
##  [541,] 0.4360510
##  [542,] 0.2206088
##  [543,] 0.5057214
##  [544,] 0.6684327
##  [545,] 0.7333883
##  [546,] 0.6648106
##  [547,] 1.1233561
##  [548,] 0.6286519
##  [549,] 0.7856403
##  [550,] 1.1472769
##  [551,] 1.3095603
##  [552,] 1.0277799
##  [553,] 0.6248721
##  [554,] 1.4135550
##  [555,] 1.1942379
##  [556,] 2.3980941
##  [557,] 1.6562564
##  [558,] 0.6749257
##  [559,] 0.4319080
##  [560,] 1.6681498
##  [561,] 1.5429487
##  [562,] 0.8937965
##  [563,] 0.7221762
##  [564,] 0.6006768
##  [565,] 1.1170719
##  [566,] 1.1476224
##  [567,] 0.5358431
##  [568,] 0.6477835
##  [569,] 0.6186261
##  [570,] 0.4164174
##  [571,] 1.1165914
##  [572,] 0.6638217
##  [573,] 0.5421480
##  [574,] 1.0820842
##  [575,] 0.9698614
##  [576,] 0.5274830
##  [577,] 1.6352787
##  [578,] 1.3082562
##  [579,] 1.5731638
##  [580,] 1.6922911
##  [581,] 0.5651807
##  [582,] 0.4870988
##  [583,] 0.5358880
##  [584,] 2.3178463
## [585,] 0.8198482
## [586,] 0.8924028
## [587,] 0.6399730
## [588,] 1.2156624
## [589,] 1.7353283
## [590,] 0.9540584
## [591,] 0.7651158
## [592,] 0.4880557
## [593,] 0.6972555
## [594,] 1.0517192
## [595,] 0.7842689
## [596,] 0.6504213
## [597,] 1.6128910
## [598,] 1.0761926
## [599,] 0.5326930
## [600,] 0.5144678
## [601,] 0.5536467
## [602,] 0.5250940
## [603,] 0.8084886
## [604,] 1.8058088
## [605,] 0.8111623
## [606,] 2.3986737
## [607,] 0.7133818
## [608,] 1.6627178
## [609,] 1.6097970
## [610,] 0.8318416
## [611,] 0.5170224
## [612,] 1.1152474
## [613,] 0.6752088
## [614,] 1.5731672
## [615,] 0.9230201
## [616,] 2.8104125
## [617,] 1.4847364
## [618,] 0.3435631
## [619,] 0.8171850
## [620,] 0.9687080
## [621,] 0.2986530
## [622,] 0.6295945
## [623,] 0.4665743
## [624,] 0.4370360
## [625,] 0.2667280
## [626,] 0.5764299
## [627,] 0.6821365
## [628,] 0.6723371
## [629,] 0.8269355
## [630,] 1.4093280
## [631,] 0.7005588
## [632,] 0.6645317
## [633,] 0.6829491
## [634,] 0.9629079
## [635,] 0.5678863
## [636,] 0.4735871
## [637,] 1.0370513
## [638,] 0.3570548
## [639,] 1.6559382
## [640,] 0.7852730
## [641,] 1.3607216
## [642,] 0.5543506
## [643,] 0.6109259
## [644,] 0.5529505
## [645,] 1.5338109
## [646,] 2.3083359
## [647,] 2.1261860
## [648,] 2.4302678
## [649,] 1.0639762
## [650,] 0.8323883
## [651,] 1.1729073
## [652,] 1.0399965
## [653,] 2.0059872
## [654,] 0.8109556
## [655,] 0.5278125
## [656,] 0.8990536
## [657,] 2.6406199
## [658,] 0.6956123
## [659,] 0.9342360
## [660,] 0.3568999
## [661,] 0.4534161
## [662,] 0.5836607
## [663,] 1.8279213
## [664,] 0.7890679
## [665,] 0.3414271
## [666,] 0.6370055
## [667,] 1.3514427
## [668,] 2.2201499
## [669,] 1.6201690
## [670,] 1.4179008
## [671,] 1.3124835
## [672,] 1.0223691
## [673,] 0.7562097
## [674,] 2.4091577
## [675,] 2.6244162
## [676,] 0.8718266
## [677,] 1.4550268
## [678,] 0.2943929
## [679,] 0.8055376
## [680,] 0.5043111
## [681,] 1.0805663
## [682,] 0.5312949
## [683,] 1.1676750
## [684,] 0.4049500
## [685,] 0.7216209
## [686,] 0.6899563
## [687,] 2.2284695
## [688,] 0.7815004
## [689,] 0.9408136
## [690,] 0.9625951
## [691,] 1.5799044
## [692,] 0.5378236
##  [693,] 1.4996070
##  [694,] 1.0452441
##  [695,] 1.0746447
##  [696,] 0.7781237
##  [697,] 0.9363130
##  [698,] 1.5547165
##  [699,] 0.7540857
##  [700,] 0.7666127
##  [701,] 0.8376216
##  [702,] 1.4408387
##  [703,] 0.3423459
##  [704,] 1.6122051
##  [705,] 0.3717113
##  [706,] 0.5536100
##  [707,] 1.2240788
##  [708,] 0.4949680
##  [709,] 0.6093240
##  [710,] 0.5506317
##  [711,] 1.3970816
##  [712,] 0.6125612
##  [713,] 1.1535572
##  [714,] 0.2848088
##  [715,] 0.8390233
##  [716,] 0.7073474
##  [717,] 0.9945054
##  [718,] 0.5106795
##  [719,] 1.8318874
##  [720,] 0.9857027
##  [721,] 1.9222191
##  [722,] 0.9814549
##  [723,] 0.7267360
##  [724,] 0.6513016
##  [725,] 2.0087815
##  [726,] 2.4569280
##  [727,] 3.7212248
##  [728,] 1.0535023
##  [729,] 0.4747193
##  [730,] 0.9209450
##  [731,] 1.4322380
##  [732,] 0.9950915
##  [733,] 1.3768059
##  [734,] 1.6047806
##  [735,] 0.5903896
##  [736,] 1.2550284
##  [737,] 1.9068539
##  [738,] 0.6470783
##  [739,] 1.8534692
##  [740,] 0.5786879
##  [741,] 1.8345386
##  [742,] 1.5221275
##  [743,] 1.8932045
##  [744,] 2.2217264
##  [745,] 1.3472773
##  [746,] 1.3586921
##  [747,] 2.2344316
##  [748,] 0.5493433
##  [749,] 1.0337467
##  [750,] 0.4788054
##  [751,] 1.2255712
##  [752,] 0.9708304
##  [753,] 0.5183156
##  [754,] 0.3824199
##  [755,] 1.2870297
##  [756,] 0.5001435
##  [757,] 0.8789035
##  [758,] 0.8862015
##  [759,] 0.4805388
##  [760,] 1.7444577
##  [761,] 0.6971863
##  [762,] 0.7014012
##  [763,] 0.7510213
##  [764,] 0.3387673
##  [765,] 1.5900700
##  [766,] 0.9391271
##  [767,] 0.3166455
##  [768,] 1.5195135
##  [769,] 0.5614094
##  [770,] 0.7073740
##  [771,] 0.7376185
##  [772,] 0.3929648
##  [773,] 1.0026366
##  [774,] 1.5355834
##  [775,] 1.1730789
##  [776,] 0.9475826
##  [777,] 0.5749952
##  [778,] 1.0708772
##  [779,] 1.3653037
##  [780,] 1.3743421
##  [781,] 0.6431378
##  [782,] 1.8291689
##  [783,] 1.6273402
##  [784,] 1.1638920
##  [785,] 0.4180676
##  [786,] 2.9628364
##  [787,] 1.4612408
##  [788,] 0.6758912
##  [789,] 0.6309021
##  [790,] 0.4436240
##  [791,] 0.9516701
##  [792,] 1.5945801
##  [793,] 1.5151046
##  [794,] 0.6000740
##  [795,] 0.4854237
##  [796,] 1.7951772
##  [797,] 0.5315087
##  [798,] 1.1008854
##  [799,] 0.9594397
##  [800,] 1.0193766
## [801,] 1.3066629
## [802,] 2.2901611
## [803,] 0.9215234
## [804,] 1.1796173
## [805,] 0.6651371
## [806,] 1.3281922
## [807,] 0.5017280
## [808,] 0.5983430
## [809,] 0.5596606
## [810,] 0.6946231
## [811,] 0.8811010
## [812,] 0.4277615
## [813,] 1.1064498
## [814,] 0.6207048
## [815,] 0.7568567
## [816,] 3.2658200
## [817,] 1.2380958
## [818,] 1.0012897
## [819,] 0.4903451
## [820,] 0.9723498
## [821,] 0.6007852
## [822,] 0.3740367
## [823,] 0.4235291
## [824,] 0.9387361
## [825,] 0.5007263
## [826,] 0.6535405
## [827,] 0.2130236
## [828,] 0.2572589
## [829,] 1.0345279
## [830,] 0.3680021
## [831,] 1.6063236
## [832,] 0.8299340
## [833,] 1.4914635
## [834,] 0.6898186
## [835,] 0.8066824
## [836,] 0.5830716
## [837,] 0.3100240
## [838,] 0.8776295
## [839,] 0.5108791
## [840,] 1.8599531
## [841,] 0.8999067
## [842,] 0.5883084
## [843,] 0.3496494
## [844,] 0.8704447
## [845,] 0.4874090
## [846,] 0.4536186
## [847,] 1.6892144
## [848,] 1.0012067
## [849,] 1.1691963
## [850,] 1.2687761
## [851,] 1.2276995
## [852,] 1.5190666
## [853,] 0.7077173
## [854,] 0.7118212
##  [855,] 0.8873393
##  [856,] 1.1227087
##  [857,] 0.8552079
##  [858,] 0.6630999
##  [859,] 0.6055535
##  [860,] 0.7241224
##  [861,] 1.2901868
##  [862,] 0.4836976
##  [863,] 0.8105928
##  [864,] 1.0017846
##  [865,] 0.9827329
##  [866,] 0.4114569
##  [867,] 1.8546833
##  [868,] 0.4508332
##  [869,] 1.6308452
##  [870,] 1.0341327
##  [871,] 1.8678902
##  [872,] 0.6921180
##  [873,] 2.0839976
##  [874,] 0.3638279
##  [875,] 0.9705961
##  [876,] 0.5960963
##  [877,] 1.7307425
##  [878,] 0.7534054
##  [879,] 1.3251386
##  [880,] 2.1276568
##  [881,] 1.0645541
##  [882,] 0.4896917
##  [883,] 2.2548179
##  [884,] 0.3078702
##  [885,] 0.5857045
##  [886,] 0.3986474
##  [887,] 0.3173673
##  [888,] 0.8898973
##  [889,] 3.5085871
##  [890,] 0.4884477
##  [891,] 1.3586533
##  [892,] 1.1770875
##  [893,] 0.6167648
##  [894,] 1.3765835
##  [895,] 1.6368104
##  [896,] 1.1916218
##  [897,] 0.9044271
##  [898,] 0.3828816
##  [899,] 0.6320279
##  [900,] 2.5283161
##  [901,] 2.2964136
##  [902,] 0.4624533
##  [903,] 0.7402569
##  [904,] 2.8571337
##  [905,] 1.0888001
##  [906,] 0.9430727
##  [907,] 1.8400394
##  [908,] 1.6052251
##  [909,] 0.5629488
##  [910,] 0.4666069
##  [911,] 0.5326146
##  [912,] 1.1009770
##  [913,] 0.3600636
##  [914,] 1.3637051
##  [915,] 1.3091147
##  [916,] 0.6675554
##  [917,] 1.5104620
##  [918,] 0.9718657
##  [919,] 1.0681407
##  [920,] 1.5767676
##  [921,] 0.5282708
##  [922,] 2.0950160
##  [923,] 3.3753982
##  [924,] 0.7932623
##  [925,] 1.3364718
##  [926,] 1.7429850
##  [927,] 1.5592848
##  [928,] 1.2429600
##  [929,] 1.0885366
##  [930,] 1.1040891
##  [931,] 0.8204032
##  [932,] 1.7375074
##  [933,] 1.0939852
##  [934,] 0.6175427
##  [935,] 0.7786988
##  [936,] 1.4666473
##  [937,] 0.8955948
##  [938,] 0.6066166
##  [939,] 0.4115048
##  [940,] 0.6173829
##  [941,] 0.8880773
##  [942,] 0.4826112
##  [943,] 0.6470810
##  [944,] 1.4858320
##  [945,] 1.2400115
##  [946,] 1.2414722
##  [947,] 1.6457560
##  [948,] 0.5304567
##  [949,] 0.5602339
##  [950,] 0.7450092
##  [951,] 0.4931953
##  [952,] 0.6166294
##  [953,] 0.4082494
##  [954,] 1.3864295
##  [955,] 0.4350476
##  [956,] 1.6834659
##  [957,] 0.6056014
##  [958,] 1.2511049
##  [959,] 0.7368266
##  [960,] 1.7376036
##  [961,] 1.0246562
##  [962,] 0.7316534
##  [963,] 0.9322262
##  [964,] 0.5368375
##  [965,] 0.4326460
##  [966,] 1.2161509
##  [967,] 0.6577529
##  [968,] 1.3338040
##  [969,] 0.3943477
##  [970,] 0.9357566
##  [971,] 1.6919491
##  [972,] 0.7736117
##  [973,] 0.6810173
##  [974,] 1.1597292
##  [975,] 0.8122390
##  [976,] 0.7016624
##  [977,] 0.9457512
##  [978,] 0.8978260
##  [979,] 1.5602790
##  [980,] 1.9453420
##  [981,] 1.6525035
##  [982,] 1.2353956
##  [983,] 1.2498147
##  [984,] 0.8919664
##  [985,] 0.5832326
##  [986,] 1.5989047
##  [987,] 0.7582047
##  [988,] 0.5835959
##  [989,] 0.7261788
##  [990,] 0.5872422
##  [991,] 0.7685625
##  [992,] 0.7814155
##  [993,] 1.4647120
##  [994,] 0.2148010
##  [995,] 0.7703010
##  [996,] 1.1334176
##  [997,] 1.7318361
##  [998,] 0.5761324
##  [999,] 1.1898241
##
## $model.matrix
##    (Intercept) quality1 quality2
## 1            1        1        0
## 2            1        1        0
## 3            1        1        0
## 4            1        1        0
## 5            1        1        0
## 6            1        1        0
## 7            1        1        0
## 8            1        1        0
## 9            1        1        0
## 10           1        1        0
## 11           1        1        0
## 12           1        1        0
## 13           1        1        0
## 14           1       -1       -1
## 15            1        -1        -1
## 16            1        -1        -1
## 17            1        -1        -1
## 18            1        -1        -1
## 19            1         0         1
## 20            1         0         1
## 21            1         0         1
## 22            1         0         1
## 23            1         0         1
## 24            1         0         1
## 25            1        -1        -1
## 26            1        -1        -1
## 27            1        -1        -1
## 28            1        -1        -1
## 29            1        -1        -1
##
## $terms
## fish ~ quality
## attr(,"variables")
## list(fish, quality)
## attr(,"factors")
##         quality
## fish          0
## quality       1
## attr(,"term.labels")
## [1] "quality"
## attr(,"order")
## [1] 1
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 1
## attr(,".Environment")
## <environment: R_GlobalEnv>
##
## attr(,"class")
## [1] "adonis"
```

```
indval <- multipatt(fish, cluster = quality, func = "IndVal.g",
                    control = how(nperm=999))
summary(indval)#shows which species are in what water quality
```

```
##
##  Multilevel pattern analysis
##  ---------------------------
##
##  Association function: IndVal.g
##  Significance level (alpha): 0.05
##
##  Total number of species: 27
##  Selected number of species: 23
##  Number of species associated to 1 group: 1
##  Number of species associated to 2 groups: 22
##
```

```
##  List of species associated to each combination:
##
##  Group MQ  #sps.  1
##       stat p.value
## Teso 0.686   0.024 *
##
##  Group HQ+MQ  #sps.  2
##       stat p.value
## Satr 0.860   0.004 **
## Phph 0.859   0.012 *
##
##  Group LQ+MQ  #sps.  20
##       stat p.value
## Alal 0.935   0.001 ***
## Gogo 0.933   0.001 ***
## Ruru 0.916   0.001 ***
## Legi 0.901   0.001 ***
## Baba 0.895   0.001 ***
## Chna 0.866   0.001 ***
## Spbi 0.866   0.001 ***
## Cyca 0.866   0.001 ***
## Acce 0.866   0.001 ***
## Lele 0.863   0.004 **
## Titi 0.853   0.006 **
## Chto 0.829   0.002 **
## Rham 0.829   0.001 ***
## Anan 0.829   0.001 ***
## Eslu 0.827   0.019 *
## Pefl 0.806   0.017 *
## Blbj 0.791   0.003 **
## Scer 0.766   0.005 **
## Abbr 0.750   0.003 **
## Icme 0.661   0.014 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```r
fish.rel <- decostand(fish, method = "total")
phi <- multipatt(fish.rel, cluster = quality, func = "r.g",
                 control = how(nperm=999))
summary(phi)#shows which species are in what water quality.
```

```
##
##  Multilevel pattern analysis
##  ---------------------------
##
##  Association function: r.g
##  Significance level (alpha): 0.05
##
##  Total number of species: 27
##  Selected number of species: 18
##  Number of species associated to 1 group: 9
##  Number of species associated to 2 groups: 9
##
##  List of species associated to each combination:
```

```
##
##  Group HQ  #sps.  3
##       stat p.value
## Phph 0.802   0.001 ***
## Neba 0.734   0.001 ***
## Satr 0.650   0.001 ***
##
##  Group LQ  #sps.  2
##       stat p.value
## Alal 0.693   0.001 ***
## Ruru 0.473   0.017 *
##
##  Group MQ  #sps.  4
##       stat p.value
## Anan 0.571   0.006 **
## Spbi 0.557   0.009 **
## Chto 0.542   0.014 *
## Icme 0.475   0.027 *
##
##  Group LQ+MQ  #sps.  9
##       stat p.value
## Legi 0.658   0.001 ***
## Baba 0.645   0.003 **
## Rham 0.600   0.007 **
## Acce 0.594   0.008 **
## Cyca 0.586   0.004 **
## Chna 0.571   0.006 **
## Blbj 0.571   0.010 **
## Gogo 0.523   0.018 *
## Abbr 0.499   0.032 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 1: Based on the PERMANOVA, IndVal, and phi coefficient analyses, what did you learn about the relationship between habitat quality and the fish species composition? Are the different analyses consistent with one another and do they agree with the visualizations (heat maps, cluster dendograms, ordinations) that you created?

> ***Answer 1***:Based on the three different analyses, water quality is important to how fish cluster together. The PERMANOVA shows that water quality is significant in shaping fish composition, while the IndVal and phi coefficient show what fish species are in what water quality. All analyses agree with the heat map in that fish cluster according to water quality. The IndVal and phi coefficient also show similar findings in what fish species are in what water quality. For example, both the IndVal and phi coefficient show that Satr and Phph are found in high quality water.

B. Multivariate Procedures for Continuous Designs

i. Mantel Test

In the R code chunk below, do the following:

1. create distance matrices for both fish communities and environmental factors, and

2. use a Mantel test to test the hypothesis that fish assemblages are correlated with stream environmental variables.

```
fish.dist <- vegdist(doubs$fish[-8,], method = "bray")
env.dist <- vegdist(scale(doubs$env[-8,]),method = "euclid")

mantel(fish.dist,env.dist)
```

```
##
## Mantel statistic based on Pearson's product-moment correlation
##
## Call:
## mantel(xdis = fish.dist, ydis = env.dist)
##
## Mantel statistic r: 0.604
##       Significance: 0.001
##
## Upper quantiles of permutations (null model):
##    90%    95%  97.5%    99%
## 0.0984 0.1472 0.1773 0.2072
## Permutation: free
## Number of permutations: 999
```

Question 2: What do the results from our Mantel test suggest about fish diversity and stream environmental conditions? How does this relate to your hypothesis about stream quality influencing fish communities?

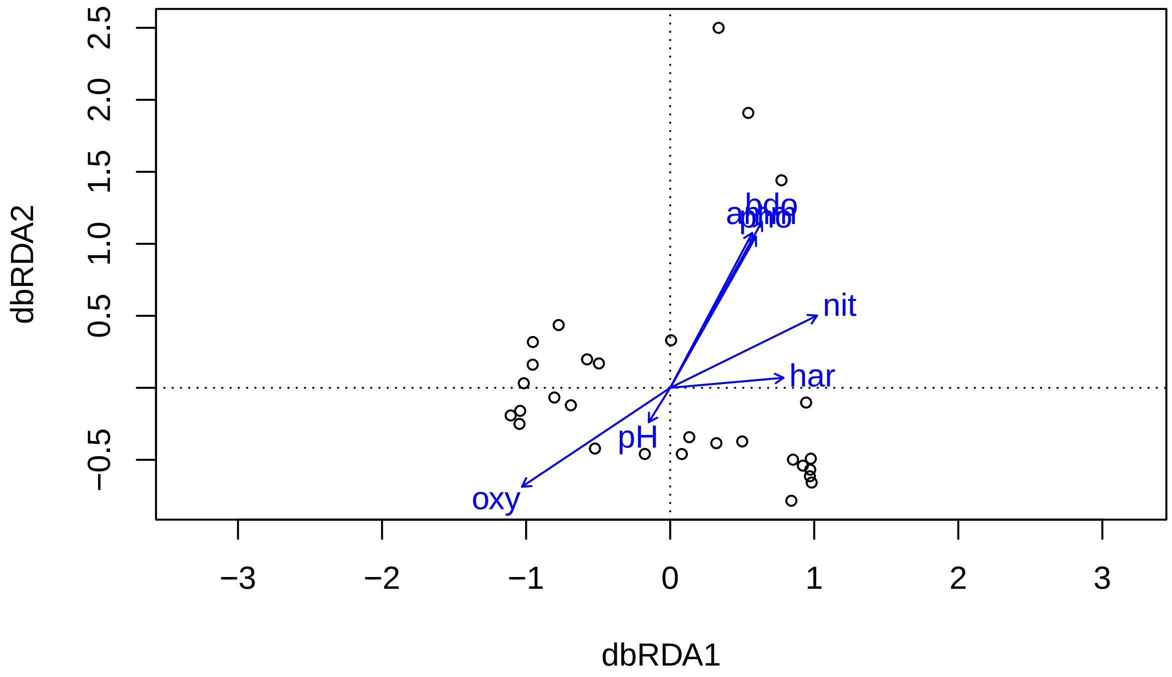
> ***Answer 2***: The Mantel test suggests that fish diversity is correlated with stream environmental conditions. This is in agreement with our prior analyses that showed water quality is significant in shaping fish community.

ii. Constrained Ordination

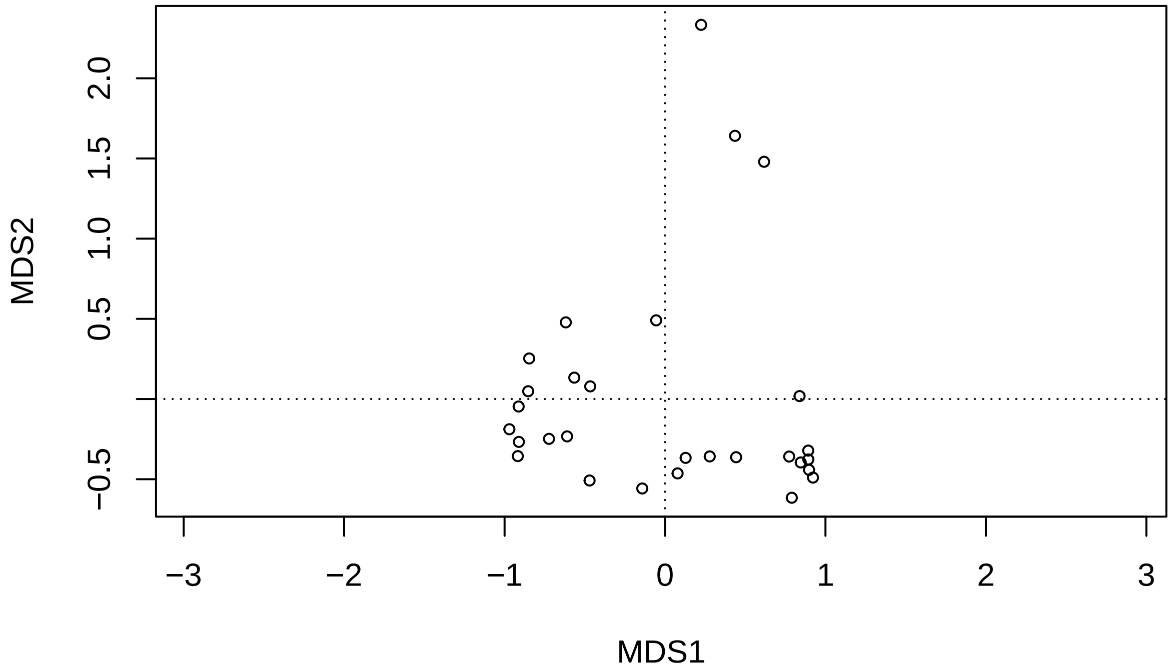
In the R code chunk below, do the following:

1. create an environmental matrix of the water chemistry data included in the `doubs` dataset using forward and reverse selection of variables,
2. conduct a redundancy analysis on the fish assemblages of the Doubs River,
3. use a permutation test to determine the significance of the constrained analysis,
4. use a permutation test to determine the correlation of each environmental factor on the constrained axes,
5. calculate the explained variation on the first and second constrained axes,
6. plot the constrained ordination results including labeled points for each site, and
7. add vectors that demonstrate the influence of each environmental factor the constrained ordination.

```
env.chem <- as.matrix(doubs$env[-8, 5:11])

fish.db <- vegdist(fish, method = "bray")
doubs.dbrda <- dbrda(fish.db ~ ., as.data.frame(env.chem))
ordiplot(doubs.dbrda)
```



```r
doubs.dbrda.mod0 <- dbrda(fish.db ~ 1, as.data.frame(env.chem))

ordiplot(doubs.dbrda.mod0)
```



```
doubs.dbrda.mod1 <- dbrda(fish.db ~ ., as.data.frame(env.chem))

doubs.dbrda <- ordiR2step(doubs.dbrda.mod0, doubs.dbrda.mod1, perm.max = 200)
```

```
## Step: R2.adj= 0
## Call: fish.db ~ 1
##
##                   R2.adjusted
## <All variables>  0.53032584
## + oxy            0.27727176
## + nit            0.25755208
## + bdo            0.17477787
## + pho            0.14568614
## + har            0.14174915
## + amm            0.14142804
## <none>           0.00000000
## + pH            -0.01827054
##
##       Df    AIC      F Pr(>F)
## + oxy  1 47.939 11.742  0.002 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.2772718
## Call: fish.db ~ oxy
```

```
##
##                  R2.adjusted
## <All variables>   0.5303258
## + bdo             0.4009000
## + amm             0.3474192
## + pho             0.3452702
## + har             0.3331357
## + nit             0.3316120
## <none>            0.2772718
## + pH              0.2586983
##
##       Df    AIC      F Pr(>F)
## + bdo  1 43.404 6.5716  0.002 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.4009
## Call: fish.db ~ oxy + bdo
##
##                  R2.adjusted
## <All variables>   0.5303258
## + nit             0.4980793
## + har             0.4695121
## <none>            0.4009000
## + pho             0.3938042
## + amm             0.3869134
## + pH              0.3865240
##
##       Df    AIC     F Pr(>F)
## + nit  1 39.134 6.034  0.004 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.4980793
## Call: fish.db ~ oxy + bdo + nit
##
##                  R2.adjusted
## + amm             0.5415705
## <All variables>   0.5303258
## + pho             0.5277128
## + har             0.5218852
## <none>            0.4980793
## + pH              0.4843267
```

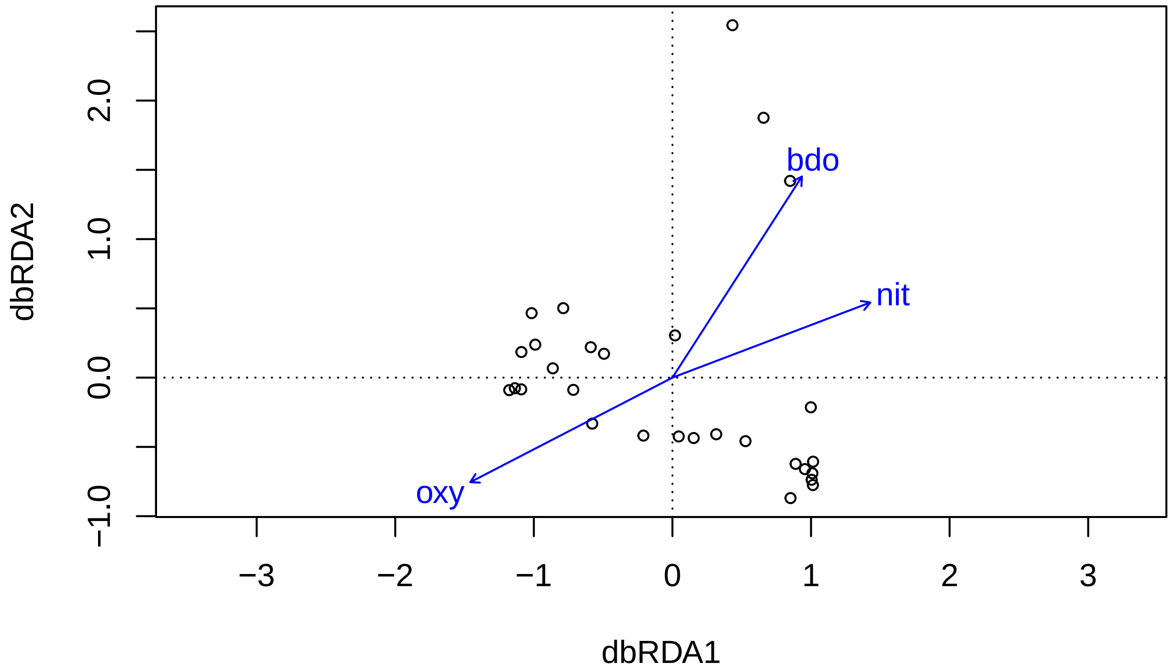
```
doubs.dbrda$call
```

```
## dbrda(formula = fish.db ~ oxy + bdo + nit, data = as.data.frame(env.chem))
```

```
doubs.dbrda$anova
```

```
##                   R2.adj Df    AIC       F Pr(>F)
## + oxy           0.27727  1 47.939 11.7421  0.002 **
## + bdo           0.40090  1 43.404  6.5716  0.002 **
## + nit            0.49808  1 39.134  6.0340  0.004 **
## <All variables> 0.53033
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ordiplot(doubs.dbrda)#shows which chem/environmental variables are the most important in influencing fi
```



```
permutest(doubs.dbrda, permutations = 999)#oxy, bdo, and nit are sign. for fish assemblage
```

```
##
## Permutation test for dbrda under reduced model
##
## Permutation: free
## Number of permutations: 999
##
## Model: dbrda(formula = fish.db ~ oxy + bdo + nit, data =
## as.data.frame(env.chem))
## Permutation test for all constrained eigenvalues
##          Df Inertia      F Pr(>F)
## Model     3  3.7317 10.262  0.001 ***
## Residual 25  3.0304
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
envfit(doubs.dbrda, env.chem[,c(4,6,7)], perm = 999)#shows how each (oxy, bdo, and nit) are sign. to in
```

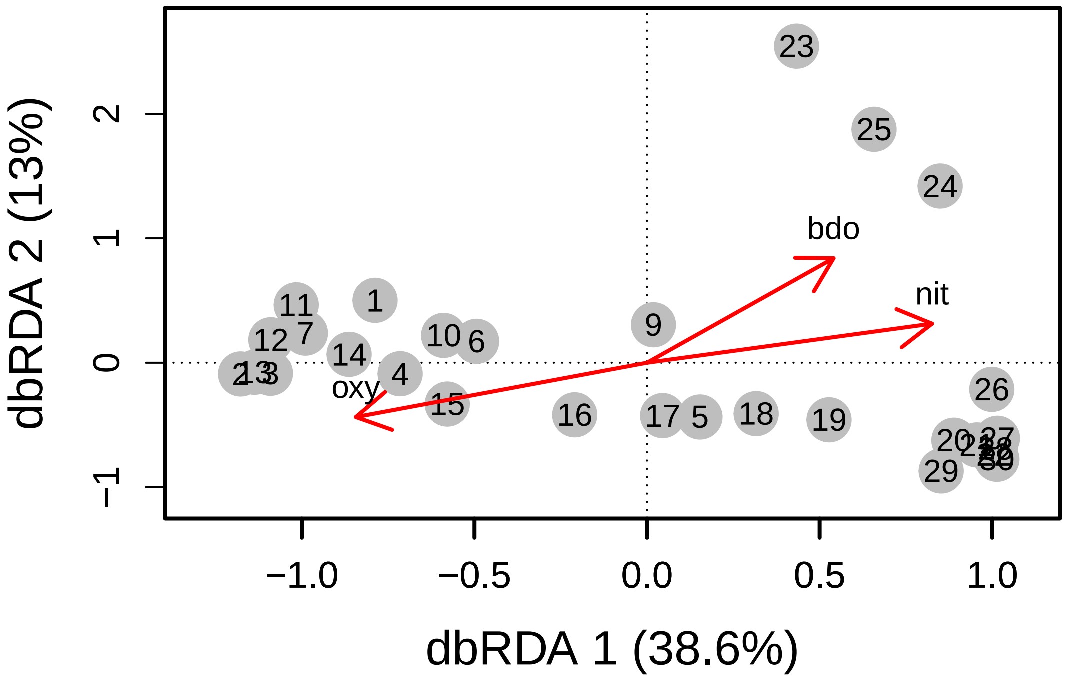
```
##
## ***VECTORS
##
##        dbRDA1   dbRDA2     r2 Pr(>r)
## nit  0.87724  0.48005 0.6431  0.001 ***
## oxy -0.82864 -0.55979 0.7656  0.001 ***
## bdo  0.55603  0.83116 0.8939  0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

```r
dbrda.explainvar1 <- round(doubs.dbrda$CCA$eig[1] /
                             sum(c(doubs.dbrda$CCA$eig, doubs.dbrda$CA$eig)), 3) * 100

dbrda.explainvar2 <- round(doubs.dbrda$CCA$eig[2] /
                             sum(c(doubs.dbrda$CCA$eig, doubs.dbrda$CA$eig)), 3) * 100

par(mar = c(5, 5, 4, 4) + 0.1)

plot(scores(doubs.dbrda, display = "wa"), xlim = c(-1.3, 1.1),
     ylim = c(-1.1, 2.7), xlab = paste("dbRDA 1 (", dbrda.explainvar1, "%)",
                                       sep = ""), ylab = paste("dbRDA 2 (", dbrda.explainvar2, "%)", se
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5,
     cex.axis = 1.2, axis = FALSE)
```

```
## Warning in plot.window(...): "axis" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "axis" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "axis" is not a
## graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "axis" is not a
## graphical parameter

## Warning in box(...): "axis" is not a graphical parameter

## Warning in title(...): "axis" is not a graphical parameter
```

```r
axis(side = 1, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

points(scores(doubs.dbrda, display = "wa"),
       pch = 19, cex = 3, bg = "gray", col = "gray")
text(scores(doubs.dbrda, display = "wa"),
     labels = row.names(scores(doubs.dbrda, display = "wa")))
```

```
vectors <- scores(doubs.dbrda, display = "bp")
arrows(0, 0, vectors[,1], vectors[,2],
       lwd = 2, lty = 1, length = 0.2, col = "red")
text(vectors[,1], vectors[,2], pos = 3,
     labels = row.names(vectors))
```



```
#axis(side = 3, lwd.ticks=2, cex.axis=1.2, las = 1, col = "red", lwd = 2.2,
     #at = pretty(range(vectors[,1])) * 2, labels = #pretty(range(vector[,1])))
#axis(side = 4, lwd.ticks=2, cexaxis=1.2, las = 1, col = "red", lwd = 2.2,
     #at = pretty(range(vectors[, 2])) * 2, labels = pretty(range(vectors[, 2])))
```

Question 3: Based on the constrained ordination, what are the environmental variables (or groups of correlated variables) that seem to be contributing to variation in fish community structure?

> ***Answer 3***: The environmental variables that seem to be contributing to variation in fish community structure are oxy, bdo, and nit. These are the most important variables in influencing fish community, with bdo explaining the greatest variation.

iii. Variation Partitioning

In the code chunk below,

1. Create a matrix model of the selected environmental variables,

2. Create a matrix model of the selected PCNM axes,
3. Perform constrained and partial constrained ordinations using the spatial and environmental models you just created,
4. Test the significance of each of your constrained ordinations using permutation tests,
5. Partition the variation among sites into the relative importance of space, environment, spatially structured environment, and residuals,
6. Plot the variation partitioning output to visualize it.

```
doubs.dbrda$anova
```

```
##                  R2.adj Df    AIC       F Pr(>F)
## + oxy           0.27727  1 47.939 11.7421  0.002 **
## + bdo           0.40090  1 43.404  6.5716  0.002 **
## + nit           0.49808  1 39.134  6.0340  0.004 **
## <All variables> 0.53033
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

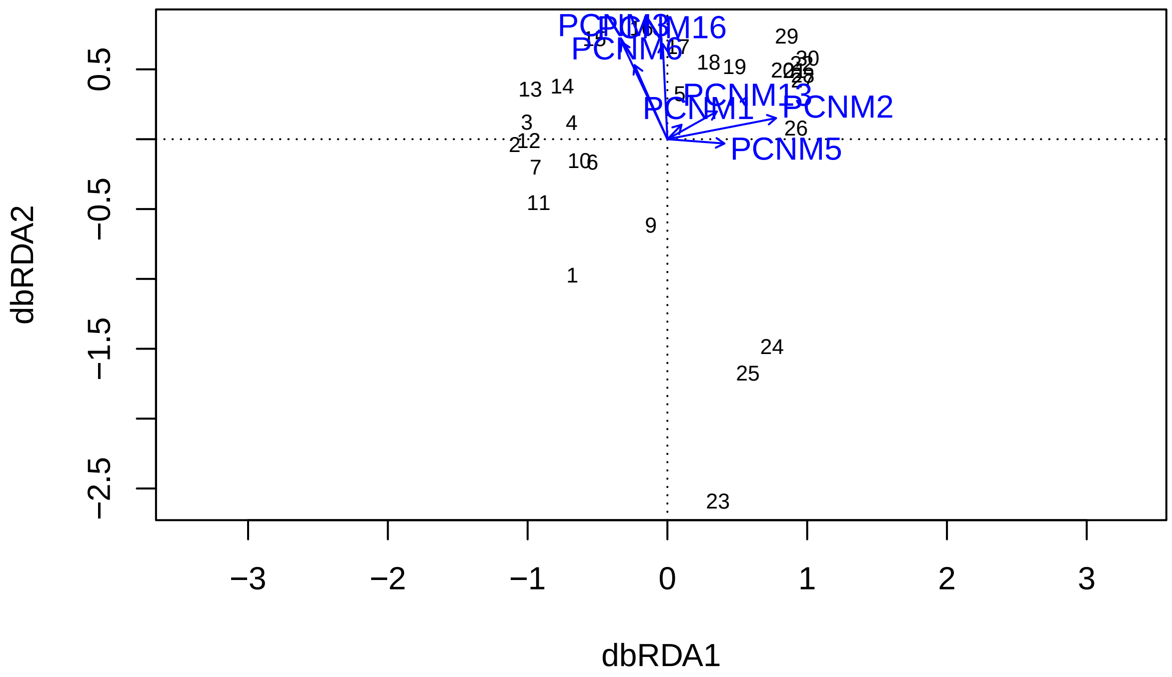
```
env.mod <- model.matrix(~ oxy + bdo + nit, as.data.frame(env.chem))[,-1]

rs <- rowSums(fish)/sum(fish)

doubs.pcnmw <- pcnm(dist(doubs$xy[-8,]), w = rs, dist.ret = T)

doubs.pcnmw$values > 0
```

```
##  [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## [13]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [25] FALSE FALSE
```

```
doubs.space <- as.data.frame(scores(doubs.pcnmw))
doubs.pcnm.mod0 <- dbrda(fish.db ~ 1, doubs.space)
doubs.pcnm.mod1 <- dbrda(fish.db ~., doubs.space)
step.pcnm <- ordiR2step(doubs.pcnm.mod0, doubs.pcnm.mod1, perm.max = 200)
```

```
## Step: R2.adj= 0
## Call: fish.db ~ 1
##
##                   R2.adjusted
## <All variables>  0.626011301
## + PCNM2          0.235370423
## + PCNM3          0.078394885
## + PCNM13         0.065305668
## + PCNM5          0.046185074
## + PCNM6          0.032809156
## + PCNM16         0.030486700
## + PCNM14         0.029680999
## + PCNM9          0.020357410
## + PCNM15         0.013632610
## + PCNM8          0.009411968
## + PCNM1          0.003986221
## + PCNM17         0.002415012
## + PCNM10         0.001326442
## <none>           0.000000000
## + PCNM7         -0.001861430
## + PCNM11        -0.006841522
## + PCNM4         -0.007089863
## + PCNM12        -0.014396973
##
##         Df    AIC     F Pr(>F)
## + PCNM2  1 49.574 9.619  0.002 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Step: R2.adj= 0.2353704
## Call: fish.db ~ PCNM2
##
##                 R2.adjusted
## <All variables>   0.6260113
## + PCNM3           0.3429270
## + PCNM5           0.3057368
## + PCNM1           0.2885396
## + PCNM16          0.2786746
## + PCNM14          0.2744520
## + PCNM15          0.2692809
## + PCNM6           0.2659866
## + PCNM13          0.2636194
## + PCNM9           0.2517847
## + PCNM8           0.2496240
## + PCNM10          0.2434688
## + PCNM7           0.2431476
## + PCNM17          0.2404343
## + PCNM11          0.2366833
## <none>            0.2353704
## + PCNM12          0.2288789
## + PCNM4           0.2189522
##
##         Df    AIC      F Pr(>F)
## + PCNM3  1 46.083 5.4196  0.002 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Step: R2.adj= 0.342927
## Call: fish.db ~ PCNM2 + PCNM3
##
##                 R2.adjusted
## <All variables>   0.6260113
## + PCNM5           0.4076020
## + PCNM1           0.3970300
## + PCNM16          0.3853210
## + PCNM15          0.3828748
## + PCNM14          0.3781827
## + PCNM13          0.3770376
## + PCNM6           0.3595644
## + PCNM8           0.3556885
## + PCNM7           0.3541631
## + PCNM10          0.3526775
## + PCNM17          0.3513683
## + PCNM9           0.3433672
## <none>            0.3429270
## + PCNM11          0.3416399
## + PCNM12          0.3396547
## + PCNM4           0.3311509
##
##         Df    AIC      F Pr(>F)
## + PCNM5  1 43.941 3.8385  0.014 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.407602
## Call: fish.db ~ PCNM2 + PCNM3 + PCNM5
##
##                 R2.adjusted
## <All variables>   0.6260113
## + PCNM1           0.4721469
## + PCNM16          0.4631976
## + PCNM15          0.4589111
## + PCNM14          0.4535248
## + PCNM13          0.4511582
## + PCNM6           0.4305640
## + PCNM7           0.4261965
## + PCNM8           0.4224505
## + PCNM17          0.4181666
## + PCNM10          0.4154485
## + PCNM11          0.4112178
## + PCNM9           0.4111995
## + PCNM12          0.4087602
## <none>            0.4076020
## + PCNM4           0.3976526
##
##         Df    AIC     F Pr(>F)
## + PCNM1  1 41.411 4.057   0.01 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.4721469
## Call: fish.db ~ PCNM2 + PCNM3 + PCNM5 + PCNM1
##
##                 R2.adjusted
## <All variables>   0.6260113
## + PCNM13          0.5212427
## + PCNM16          0.5208668
## + PCNM15          0.5161770
## + PCNM14          0.5147355
## + PCNM6           0.4999020
## + PCNM7           0.4936559
## + PCNM8           0.4904113
## + PCNM17          0.4856884
## + PCNM10          0.4835952
## + PCNM11          0.4760087
## + PCNM9           0.4751424
## + PCNM12          0.4747221
## <none>            0.4721469
## + PCNM4           0.4651218
##
##          Df    AIC      F Pr(>F)
## + PCNM13  1 39.346 3.4612  0.024 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.5212427
## Call: fish.db ~ PCNM2 + PCNM3 + PCNM5 + PCNM1 + PCNM13
##
##                  R2.adjusted
## <All variables>   0.6260113
## + PCNM16          0.5767968
## + PCNM15          0.5715331
## + PCNM14          0.5698343
## + PCNM6           0.5475140
## + PCNM7           0.5392074
## + PCNM8           0.5379134
## + PCNM11          0.5281106
## + PCNM9           0.5267003
## + PCNM10          0.5265029
## + PCNM12          0.5255581
## <none>            0.5212427
## + PCNM17          0.5171800
## + PCNM4           0.5152311
##
##          Df   AIC      F Pr(>F)
## + PCNM16  1 36.48 4.0192   0.01 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.5767968
## Call: fish.db ~ PCNM2 + PCNM3 + PCNM5 + PCNM1 + PCNM13 + PCNM16
##
##                  R2.adjusted
## <All variables>   0.6260113
## + PCNM6           0.6043089
## + PCNM8           0.5970286
## + PCNM12          0.5946888
## + PCNM7           0.5946475
## + PCNM9           0.5883735
## + PCNM10          0.5851333
## + PCNM15          0.5846468
## <none>            0.5767968
## + PCNM17          0.5748533
## + PCNM4           0.5733749
## + PCNM11          0.5711176
## + PCNM14          0.5652509
##
##         Df    AIC      F Pr(>F)
## + PCNM6  1 35.182 2.5296  0.044 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step: R2.adj= 0.6043089
## Call: fish.db ~ PCNM2 + PCNM3 + PCNM5 + PCNM1 + PCNM13 + PCNM16 + PCNM6
##
##                   R2.adjusted
## <All variables>     0.6260113
## + PCNM8             0.6248697
## + PCNM12            0.6208788
## + PCNM10            0.6170988
## + PCNM7             0.6142419
## + PCNM15            0.6140369
## + PCNM9             0.6107110
## <none>              0.6043089
## + PCNM17            0.6037430
## + PCNM11            0.5978305
## + PCNM4             0.5963667
## + PCNM14            0.5932113
##
##          Df    AIC     F Pr(>F)
## + PCNM8  1 34.219 2.151  0.088 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(step.pcnm)
```



```
step.pcnm$anova
```

```
##                    R2.adj Df    AIC      F Pr(>F)
## + PCNM2           0.23537  1 49.574 9.6190  0.002 **
## + PCNM3           0.34293  1 46.083 5.4196  0.002 **
## + PCNM5           0.40760  1 43.941 3.8385  0.014 *
## + PCNM1           0.47215  1 41.411 4.0570  0.010 **
## + PCNM13          0.52124  1 39.346 3.4612  0.024 *
## + PCNM16          0.57680  1 36.480 4.0192  0.010 **
## + PCNM6           0.60431  1 35.182 2.5296  0.044 *
## <All variables> 0.62601
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
space.mod <- model.matrix(~ PCNM2 + PCNM3 + PCNM5 + PCNM1 +
                            PCNM13 + PCNM16 + PCNM6, doubs.space)[,-1]

doubs.total.env <- dbrda(fish.db ~ env.mod)
doubs.total.space <- dbrda(fish.db ~ space.mod)

doubs.env.cond.space <- dbrda(fish.db ~ env.mod + Condition(space.mod))
doubs.space.cond.env <- dbrda(fish.db ~ space.mod + Condition(env.mod))

permutest(doubs.env.cond.space, permutations = 999)
```

```
##
## Permutation test for dbrda under reduced model
##
## Permutation: free
## Number of permutations: 999
##
## Model: dbrda(formula = fish.db ~ env.mod + Condition(space.mod))
## Permutation test for all constrained eigenvalues
##          Df Inertia     F Pr(>F)
## Model     3 0.85158 4.423  0.001 ***
## Residual 18 1.15519
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
permutest(doubs.space.cond.env, permutations = 999)
```

```
##
## Permutation test for dbrda under reduced model
##
## Permutation: free
## Number of permutations: 999
##
## Model: dbrda(formula = fish.db ~ space.mod + Condition(env.mod))
## Permutation test for all constrained eigenvalues
##          Df Inertia      F Pr(>F)
## Model     7  1.8752 4.1741  0.001 ***
## Residual 18  1.1552
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
permutest(doubs.total.env, permutations = 999)
```

```
##
## Permutation test for dbrda under reduced model
##
## Permutation: free
## Number of permutations: 999
##
## Model: dbrda(formula = fish.db ~ env.mod)
## Permutation test for all constrained eigenvalues
##          Df Inertia      F Pr(>F)
## Model     3  3.7317 10.262  0.001 ***
## Residual 25  3.0304
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
permutest(doubs.total.space, permutations = 999)
```

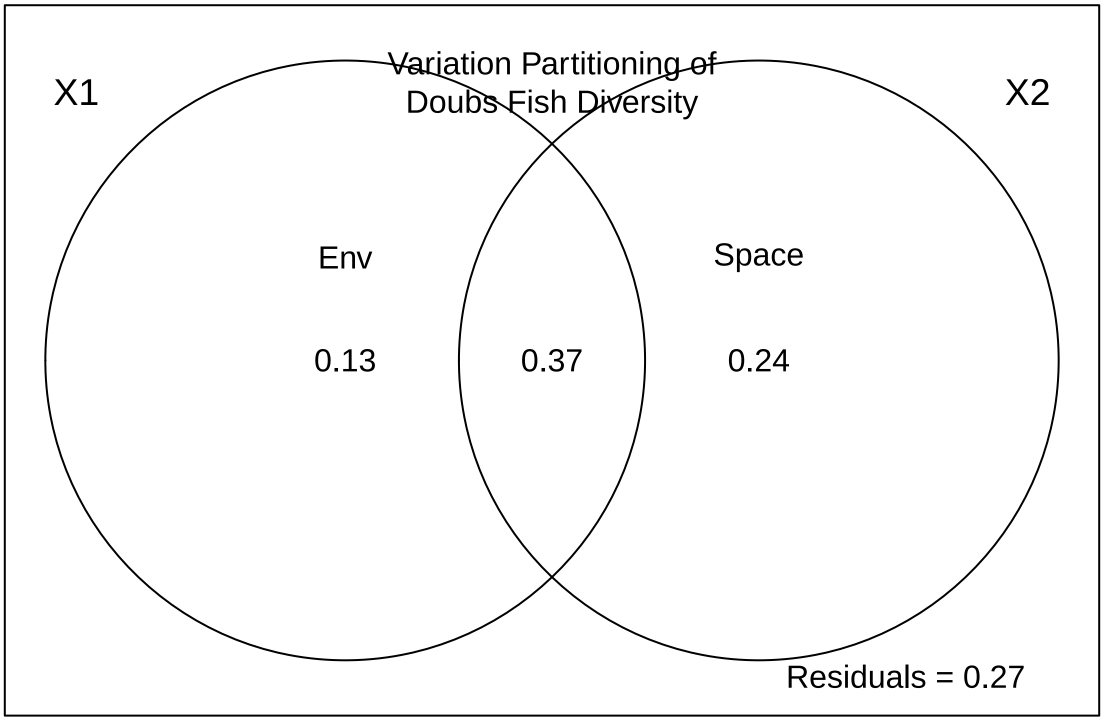
```
##
## Permutation test for dbrda under reduced model
##
## Permutation: free
## Number of permutations: 999
##
## Model: dbrda(formula = fish.db ~ space.mod)
## Permutation test for all constrained eigenvalues
##          Df Inertia      F Pr(>F)
## Model     7  4.7553 7.1089  0.001 ***
## Residual 21  2.0068
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
doubs.varpart <- varpart(fish.db, env.mod, space.mod)
doubs.varpart
```

```
##
## Partition of squared Bray distance in dbRDA
##
## Call: varpart(Y = fish.db, X = env.mod, space.mod)
##
## Explanatory tables:
## X1:  env.mod
## X2:  space.mod
##
## No. of explanatory tables: 2
## Total variation (SS): 6.7621
## No. of observations: 29
##
```

```
## Partition table:
##                      Df R.squared Adj.R.squared Testable
## [a+c] = X1            3   0.55186       0.49808     TRUE
## [b+c] = X2            7   0.70323       0.60431     TRUE
## [a+b+c] = X1+X2      10   0.82917       0.73426     TRUE
## Individual fractions
## [a] = X1|X2           3                 0.12995     TRUE
## [b] = X2|X1           7                 0.23618     TRUE
## [c]                   0                 0.36813    FALSE
## [d] = Residuals                         0.26574    FALSE
## ---
## Use function 'dbrda' to test significance of fractions of interest
```

```
par(mar = c(2,2,2,2))
plot(doubs.varpart)
text(1, 0.25, "Space")
text(0, 0.25, "Env")
mtext("Variation Partitioning of\nDoubs Fish Diversity", side = 3, line = -3)
```



Question 4: Interpret the variation partitioning results.

> ***Answer 4***: The variation partitioning results show us that space explains more of the variation in fish community structure than environment does. However, when space and environment are coupled together they explain even more of the variation in fish community.

SYNTHESIS

Load the dataset from that you and your partner are using for the team project. Use one of the hypothesis-testing tools introduced in the beta diversity module. Interpret the findings of your data with respect to principles of biodiversity.

```
install.packages("tidyverse")
```

```
## Installing package into ’C:/Users/jonat/AppData/Local/R/win-library/4.2’
## (as ’lib’ is unspecified)
```

```
## package ’tidyverse’ successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\jonat\AppData\Local\Temp\RtmpGAuk1w\downloaded_packages
```

```
library(readr)
library(ggplot2)
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## Error: package or namespace load failed for ’tidyverse’ in loadNamespace(j <- i[[1L]], c(lib.loc, .li
##  namespace ’cli’ 3.3.0 is already loaded, but >= 3.4.0 is required
```

```
library(plyr)
#import data
zoopfield <- read.csv("C:\\Users\\jonat\\OneDrive\\Documents\\Quantitative Biodiversity\\zoopfield_2009
#check structure and change lake names to numeric values
str(zoopfield)
```

```
## ’data.frame’:    282 obs. of  88 variables:
##  $ Lake_Name     : chr  "Airline" "Airline" "Airline" "Airline" ...
##  $ Round         : int  1 2 4 5 6 7 8 9 10 11 ...
##  $ date          : chr  "8/21/2009" "8/25/2009" "9/8/2009" "9/15/2009" ...
##  $ JD            : int  233 237 251 258 265 272 280 286 293 300 ...
##  $ M_m_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_n_l         : int  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ M_m_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_s_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_n_e         : int  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ U_m_l         : num  1.31 1.26 NA 1.29 1.26 1.24 1.28 NA 1.24 1.29 ...
##  $ U_s_l         : num  0.1 0.1 NA 0.1 0.11 0.08 0.1 NA 0.06 0.09 ...
##  $ U_n_l         : int  31 40 NA 23 28 29 25 NA 26 31 ...
##  $ U_m_e         : num  1.06 0.75 NA 0.78 0.93 0.59 1.68 NA 0.88 2.1 ...
##  $ U_s_e         : num  0.57 0.54 NA 0.74 0.38 0.78 0.8 NA 0.77 1.01 ...
##  $ U_n_e         : int  31 40 NA 23 28 29 25 NA 26 31 ...
##  $ O_m_l         : num  NA NA NA NA NA 1.24 NA NA NA NA ...
##  $ O_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ O_n_l         : int  0 0 NA 0 0 1 0 NA 0 0 ...
## $ O_m_e         : num  NA NA NA NA NA NA NA NA NA NA ...
## $ O_s_e         : num  NA NA NA NA NA NA NA NA NA NA ...
## $ O_n_e         : int  0 0 NA 0 0 1 0 NA 0 0 ...
## $ E_m_l         : num  NA NA NA NA NA NA NA NA NA NA ...
## $ E_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
## $ E_n_l         : int  0 0 NA 0 0 0 0 NA 0 0 ...
## $ E_Ratio       : num  1.06 0.75 NA 0.75 0.93 0.59 1.68 NA 0.88 2.1 ...
## $ M_m_l2        : logi  NA NA NA NA NA NA ...
## $ M_s_l2        : logi  NA NA NA NA NA NA ...
## $ U_m_l2        : logi  NA NA NA NA NA NA ...
## $ U_s_l2        : logi  NA NA NA NA NA NA ...
## $ O_m_l2        : logi  NA NA NA NA NA NA ...
## $ O_s_l2        : logi  NA NA NA NA NA NA ...
## $ E_m_l2        : logi  NA NA NA NA NA NA ...
## $ E_s_l2        : logi  NA NA NA NA NA NA ...
## $ Total_Chl     : num  2.15 2.29 2.2 2.5 2.69 3.57 3.9 NA 3.86 3.83 ...
## $ Edible_Chl    : num  1.55 1.58 1.82 1.61 1.86 3.01 3.25 NA 3.42 3.27 ...
## $ dent          : chr  "62,436.84" "118,327.89" "200,150.36" "39,442.63" ...
## $ cerio         : chr  "1,007.05" "2,517.61" "8,811.65" "1,678.41" ...
## $ pulic         : chr  "114,299.70" "275,678.80" "281,972.84" "228,263.73" ...
## $ Bosmina       : chr  "0" "0" "0" "839.2" ...
## $ parv          : chr  "0" "0" "0" "0" ...
## $ ambig         : chr  "0" "0" "0" "0" ...
## $ diaph         : chr  "0" "0" "0" "0" ...
## $ scaph         : chr  "0" "0" "0" "0" ...
## $ alona         : chr  "0" "0" "0" "0" ...
## $ chaob         : int  31 19 82 52 33 15 24 2 1 0 ...
## $ ostr          : logi  NA NA NA NA NA NA ...
## $ cal           : logi  NA NA NA NA NA NA ...
## $ cyc           : logi  NA NA NA NA NA NA ...
## $ overall_m_prev: num  0 0 NA 0.07 0 0.01 0.01 NA 0 0 ...
## $ j_m_prev      : num  NA NA NA 0.76 NA 0 0 NA NA NA ...
## $ a_m_prev      : num  NA NA NA 0.24 NA 1 1 NA NA NA ...
## $ m_m_prev      : num  NA NA NA 0 NA 0 0 NA NA NA ...
## $ e_m_prev      : num  NA NA NA 0 NA 0 0 NA NA NA ...
## $ s_prev        : num  0.02 0.01 NA 0 0.02 0.04 0 NA 0.02 0 ...
## $ p_prev        : logi  NA NA NA NA NA NA ...
## $ pc_prev       : num  0 0.01 NA 0 0.01 0.01 0 NA 0 0 ...
## $ bc_prev       : num  0 0 NA 0 0 0 0 NA 0 0 ...
## $ sp_prev       : num  0 0 NA 0 0 0 0 NA 0 0 ...
## $ prev_count    : chr  "249" "357" "" "247" ...
## $ juv_freq      : num  0.37 0.42 NA 0.65 0.62 0.58 0.26 NA 0.61 0.37 ...
## $ male_freq     : num  0 0 NA 0 0 0 0.02 NA 0.02 0 ...
## $ ephip_freq    : num  0 0 NA 0 0 0 0 NA 0 0 ...
## $ adult_freq    : num  0.63 0.58 NA 0.35 0.38 0.42 0.72 NA 0.37 0.63 ...
## $ N_ug          : num  250 238 244 244 280 ...
## $ P_ug          : num  5.34 6.54 4.47 4.85 7.64 ...
## $ P_ug.1        : logi  NA NA NA NA NA NA ...
## $ C_ug          : logi  NA NA NA NA NA NA ...
## $ P_umol        : logi  NA NA NA NA NA NA ...
## $ C_umol        : logi  NA NA NA NA NA NA ...
## $ C_P_ratio     : logi  NA NA NA NA NA NA ...
## $ Spore_yield   : logi  NA NA NA NA NA NA ...
## $ Z_bottom      : num  12.4 16 12 12.1 10.9 ...
##  $ n_max         : num  54.1 59.2 58.4 52.4 52.8 ...
##  $ bfs           : num  0.09 0.1 0.1 0.09 0.09 0.08 0.07 0.05 0.04 0.03 ...
##  $ d_tcline      : num  6.35 5.68 6.59 6.61 6.64 ...
##  $ t_tcline      : num  21.4 24.1 21.5 22.2 22.1 ...
##  $ depth_Ezi     : num  4.25 5.08 5.49 6.01 6.09 ...
##  $ T_Ezi         : num  27.7 26.1 24.3 23.8 23.6 ...
##  $ d1mgL         : num  12.4 12.6 12 12.1 10.9 ...
##  $ mDO           : num  7.09 7.7 8.06 8.3 8.16 7.85 7.45 7.61 6.65 6.99 ...
##  $ mspc          : chr  "1,378.60" "1,391.00" "1,406.00" "1,413.50" ...
##  $ mpH           : num  7.49 7.59 7.53 7.57 0.83 7.33 7.28 7.19 6.87 7.11 ...
##  $ mT            : num  28 26.3 24.7 24.6 24 ...
##  $ mchlv         : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.02 0.01 0.02 ...
##  $ mchla         : num  0.63 0.53 0.62 0.68 0.81 0.92 0.93 1.23 1.07 1.51 ...
##  $ wT            : logi  NA NA NA NA NA NA ...
##  $ wAT           : logi  NA NA NA NA NA NA ...
```

```
zoopfield$Lake_Name <- revalue(zoopfield$Lake_Name, c("Airline" ="1", "Beaver Dam" ="2", "Beaver dam" =
```

```
## The following ‘from‘ values were not present in ‘x‘: Beaver dam, T-lake, university, Willow, Wampler
```

```
zoopfield$Lake_Name<- as.numeric(as.character(zoopfield$Lake_Name))
str(zoopfield)
```

```
## 'data.frame':    282 obs. of  88 variables:
##  $ Lake_Name     : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Round         : int  1 2 4 5 6 7 8 9 10 11 ...
##  $ date          : chr  "8/21/2009" "8/25/2009" "9/8/2009" "9/15/2009" ...
##  $ JD            : int  233 237 251 258 265 272 280 286 293 300 ...
##  $ M_m_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_n_l         : int  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ M_m_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_s_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ M_n_e         : int  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ U_m_l         : num  1.31 1.26 NA 1.29 1.26 1.24 1.28 NA 1.24 1.29 ...
##  $ U_s_l         : num  0.1 0.1 NA 0.1 0.11 0.08 0.1 NA 0.06 0.09 ...
##  $ U_n_l         : int  31 40 NA 23 28 29 25 NA 26 31 ...
##  $ U_m_e         : num  1.06 0.75 NA 0.78 0.93 0.59 1.68 NA 0.88 2.1 ...
##  $ U_s_e         : num  0.57 0.54 NA 0.74 0.38 0.78 0.8 NA 0.77 1.01 ...
##  $ U_n_e         : int  31 40 NA 23 28 29 25 NA 26 31 ...
##  $ O_m_l         : num  NA NA NA NA NA 1.24 NA NA NA NA ...
##  $ O_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ O_n_l         : int  0 0 NA 0 0 1 0 NA 0 0 ...
##  $ O_m_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ O_s_e         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ O_n_e         : int  0 0 NA 0 0 1 0 NA 0 0 ...
##  $ E_m_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ E_s_l         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ E_n_l         : int  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ E_Ratio       : num  1.06 0.75 NA 0.75 0.93 0.59 1.68 NA 0.88 2.1 ...
##  $ M_m_l2        : logi  NA NA NA NA NA NA ...
##  $ M_s_l2        : logi  NA NA NA NA NA NA ...
##  $ U_m_l2        : logi  NA NA NA NA NA NA ...
##  $ U_s_l2        : logi  NA NA NA NA NA NA ...
##  $ O_m_l2        : logi  NA NA NA NA NA NA ...
##  $ O_s_l2        : logi  NA NA NA NA NA NA ...
##  $ E_m_l2        : logi  NA NA NA NA NA NA ...
##  $ E_s_l2        : logi  NA NA NA NA NA NA ...
##  $ Total_Chl     : num  2.15 2.29 2.2 2.5 2.69 3.57 3.9 NA 3.86 3.83 ...
##  $ Edible_Chl    : num  1.55 1.58 1.82 1.61 1.86 3.01 3.25 NA 3.42 3.27 ...
##  $ dent          : chr  "62,436.84" "118,327.89" "200,150.36" "39,442.63" ...
##  $ cerio         : chr  "1,007.05" "2,517.61" "8,811.65" "1,678.41" ...
##  $ pulic         : chr  "114,299.70" "275,678.80" "281,972.84" "228,263.73" ...
##  $ Bosmina       : chr  "0" "0" "0" "839.2" ...
##  $ parv          : chr  "0" "0" "0" "0" ...
##  $ ambig         : chr  "0" "0" "0" "0" ...
##  $ diaph         : chr  "0" "0" "0" "0" ...
##  $ scaph         : chr  "0" "0" "0" "0" ...
##  $ alona         : chr  "0" "0" "0" "0" ...
##  $ chaob         : int  31 19 82 52 33 15 24 2 1 0 ...
##  $ ostr          : logi  NA NA NA NA NA NA ...
##  $ cal           : logi  NA NA NA NA NA NA ...
##  $ cyc           : logi  NA NA NA NA NA NA ...
##  $ overall_m_prev: num  0 0 NA 0.07 0 0.01 0.01 NA 0 0 ...
##  $ j_m_prev      : num  NA NA NA 0.76 NA 0 0 NA NA NA ...
##  $ a_m_prev      : num  NA NA NA 0.24 NA 1 1 NA NA NA ...
##  $ m_m_prev      : num  NA NA NA 0 NA 0 0 NA NA NA ...
##  $ e_m_prev      : num  NA NA NA 0 NA 0 0 NA NA NA ...
##  $ s_prev        : num  0.02 0.01 NA 0 0.02 0.04 0 NA 0.02 0 ...
##  $ p_prev        : logi  NA NA NA NA NA NA ...
##  $ pc_prev       : num  0 0.01 NA 0 0.01 0.01 0 NA 0 0 ...
##  $ bc_prev       : num  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ sp_prev       : num  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ prev_count    : chr  "249" "357" "" "247" ...
##  $ juv_freq      : num  0.37 0.42 NA 0.65 0.62 0.58 0.26 NA 0.61 0.37 ...
##  $ male_freq     : num  0 0 NA 0 0 0 0.02 NA 0.02 0 ...
##  $ ephip_freq    : num  0 0 NA 0 0 0 0 NA 0 0 ...
##  $ adult_freq    : num  0.63 0.58 NA 0.35 0.38 0.42 0.72 NA 0.37 0.63 ...
##  $ N_ug          : num  250 238 244 244 280 ...
##  $ P_ug          : num  5.34 6.54 4.47 4.85 7.64 ...
##  $ P_ug.1        : logi  NA NA NA NA NA NA ...
##  $ C_ug          : logi  NA NA NA NA NA NA ...
##  $ P_umol        : logi  NA NA NA NA NA NA ...
##  $ C_umol        : logi  NA NA NA NA NA NA ...
##  $ C_P_ratio     : logi  NA NA NA NA NA NA ...
##  $ Spore_yield   : logi  NA NA NA NA NA NA ...
##  $ Z_bottom      : num  12.4 16 12 12.1 10.9 ...
##  $ n_max         : num  54.1 59.2 58.4 52.4 52.8 ...
##  $ bfs           : num  0.09 0.1 0.1 0.09 0.09 0.08 0.07 0.05 0.04 0.03 ...
##  $ d_tcline      : num  6.35 5.68 6.59 6.61 6.64 ...
##  $ t_tcline      : num  21.4 24.1 21.5 22.2 22.1 ...
##  $ depth_Ezi     : num  4.25 5.08 5.49 6.01 6.09 ...
##  $ T_Ezi         : num  27.7 26.1 24.3 23.8 23.6 ...
##  $ d1mgL         : num  12.4 12.6 12 12.1 10.9 ...
##  $ mDO           : num  7.09 7.7 8.06 8.3 8.16 7.85 7.45 7.61 6.65 6.99 ...
##  $ mspc          : chr  "1,378.60" "1,391.00" "1,406.00" "1,413.50" ...
##  $ mpH           : num  7.49 7.59 7.53 7.57 0.83 7.33 7.28 7.19 6.87 7.11 ...
##  $ mT            : num  28 26.3 24.7 24.6 24 ...
##  $ mchlv         : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.02 0.01 0.02 ...
##  $ mchla         : num  0.63 0.53 0.62 0.68 0.81 0.92 0.93 1.23 1.07 1.51 ...
##  $ wT            : logi  NA NA NA NA NA NA ...
##  $ wAT           : logi  NA NA NA NA NA NA ...
```

```
#subsetting data into a data frame with lake name, zooplankton species, and environmental characteristi
zoopwork <- zoopfield[, c(1, 37:49, 73:88)]
#aggregating all rounds for a lake (taking sum of species and environmental characteristics)
library(dplyr)
```

```
##
## Attaching package: ’dplyr’

## The following objects are masked from ’package:plyr’:
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from ’package:stats’:
##
##     filter, lag

## The following objects are masked from ’package:base’:
##
##     intersect, setdiff, setequal, union
```

```
#zoopagg <- zoopwork %>%
  #group_by(Lake_Name)%>%
  #summarize_all(sum, na.rm = TRUE)
#creating vectors for the factors to test hypothesis
#species <- as.matrix(zoopagg[, c(2:14)])
#env.lakes <- as.matrix(zoopagg[, c(15:30)])
library(vegan)
library(ade4)
#PERMANOVA
#adonis2(species ~ env.lakes, method = "bray", permutations = 999)
#bray-curtis dissimilarity
#zoop.db <- vegdist(species, method = "bray", na.rm = TRUE)
#CONSTRAINED ORDINATION PLOT
#zoop.dbrda <- dbrda(zoop.db ~ ., as.data.frame(env.lakes))
#ordiplot(zoop.dbrda)
#model only the intercept
#zoop.dbrda.mod0 <- dbrda(zoop.db ~ 1, as.data.frame(env.lakes))
#ordiplot(zoop.dbrda.mod0)
#model full model, with all explanatory variables
#zoop.dbrda.mod1 <- dbrda(zoop.db ~ ., as.data.frame(env.lakes))
#all combinations of explanatory variables in model-- function returns model with lowest AIC value
#zoop.dbrda <- ordiR2step(zoop.dbrda.mod0, zoop.dbrda.mod1, perm.max = 200)
#which model was selected
#zoop.dbrda$call
#zoop.dbrda$anova
```

```
#ordiplot(zoop.dbrda)
#calculate explained variation
#dbrda.explainvar1 <- round(zoop.dbrda$CCA$eig[1] / sum(c(zoop.dbrda$CCA$eig, zoop.dbrda$CA$eig)), 3) *
#dbrda.explainvar2 <- round(zoop.dbrda$CCA$eig[2] / sum(c(zoop.dbrda$CCA$eig, zoop.dbrda$CA$eig)), 3) *
#define plot parameters
#par(mar = c(5, 5, 4, 4) + 0.1)
#initiate plot
#plot(scores(zoop.dbrda, display = "wa"), xlim = c(-2.0, 2.0),
     #ylim = c(-2.0, 2.0), xlab = paste("dbRDA 1 (", dbrda.explainvar1, "%)",
                                      # sep = ""), ylab = paste("dbRDA 2 (", #dbrda.explainvar2, "%)",
     #pch = 16, cex = 2.0, type = "n", cex.lab = 1.5, cex.axis = 1.2, axes = FALSE)
#add axes
#axis(side = 1, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
#axis(side = 2, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
#abline(h = 0, v = 0, lty = 3)
#box(lwd = 2)
#add points and labels
#points(scores(zoop.dbrda, display = "wa"),
       #pch = 19, cex = 3, bg = "gray", col = "gray")
#text(scores(zoop.dbrda, display = "wa"),
     #labels = row.names(scores(zoop.dbrda, display = "wa")))
#add environmental vectors
#vectors <- scores(zoop.dbrda, display = "bp")
#row.names(vectors) <- rownames(vectors)
#arrows(0, 0, vectors[,1], vectors[,2],
       #lwd = 2, lty = 1, length = 0.2, col = "red")
#text(vectors[,1], vectors[,2], pos = 3,
     #labels = row.names(vectors))
#axis(side = 3, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", lwd = 2.2,
     #at = pretty(range(vectors[,1]))* 2, labels = pretty(range(vectors[, 1])))
#axis(side = 4, lwd.ticks = 2, cex.axis = 1.2, las = 1, col = "red", lwd = 2.2,
     #at = pretty(range(vectors[, 2])) * 2, labels = pretty(range(vectors[, 2])))
```