homework_5

Jasmine Williamson

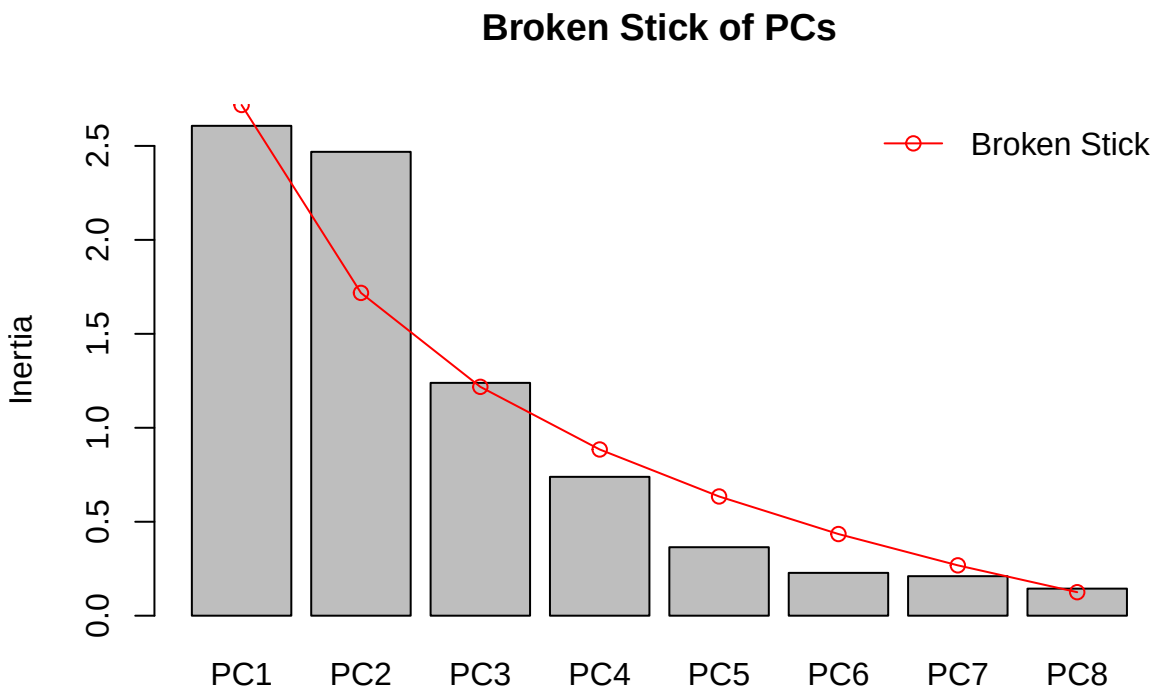
2024-10-28

Homework 5

Ordination

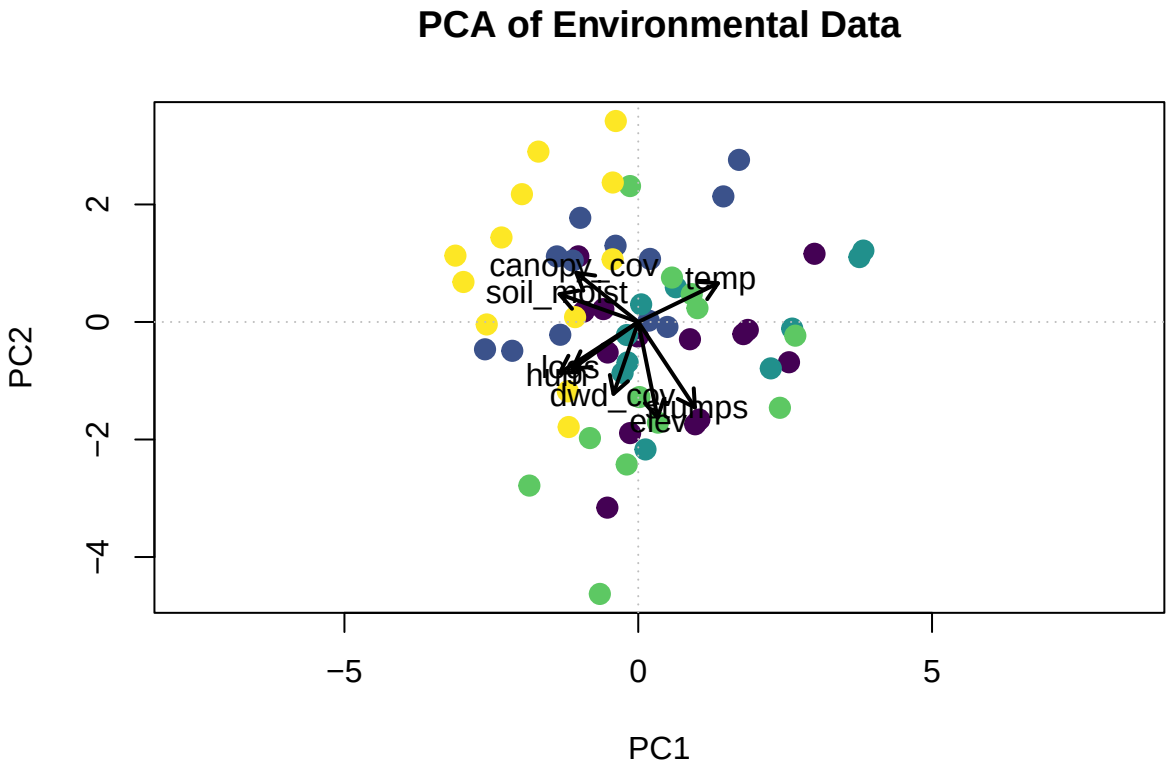
Question 1) Conduct a Principal Component Analysis (PCA), Correspondence Analysis (CA), Principal Coordinate Analysis (PCoA), and Nonmetric Multidimensional Scaling Analysis (NMDS) using your dataset.

PCA - Environmental data




```
##
## Eigenvectors:
```

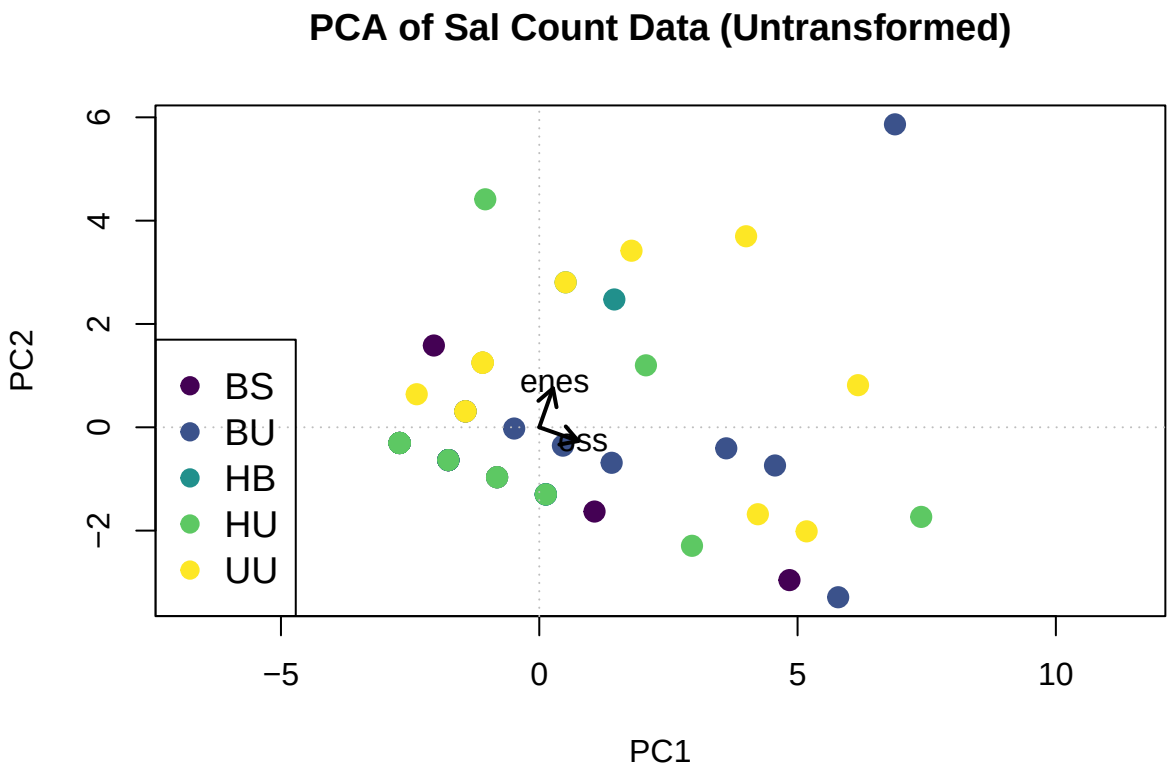
```
##                PC1    PC2    PC3    PC4    PC5    PC6
## elev         0.110 -0.542         -0.396  0.326  0.516
## temp         0.452  0.220 -0.390  0.225  0.224  0.114
## hum         -0.445 -0.289  0.342         -0.112 -0.342
## canopy_cov -0.352  0.279 -0.299 -0.648  0.318
## dwd_cov    -0.140 -0.408 -0.565  0.183  0.349 -0.526
## soil_moist -0.446  0.158  0.230  0.528  0.582  0.318
## stumps      0.322 -0.485  0.197  0.101  0.154
## logs       -0.373 -0.265 -0.477  0.219 -0.495  0.475
```



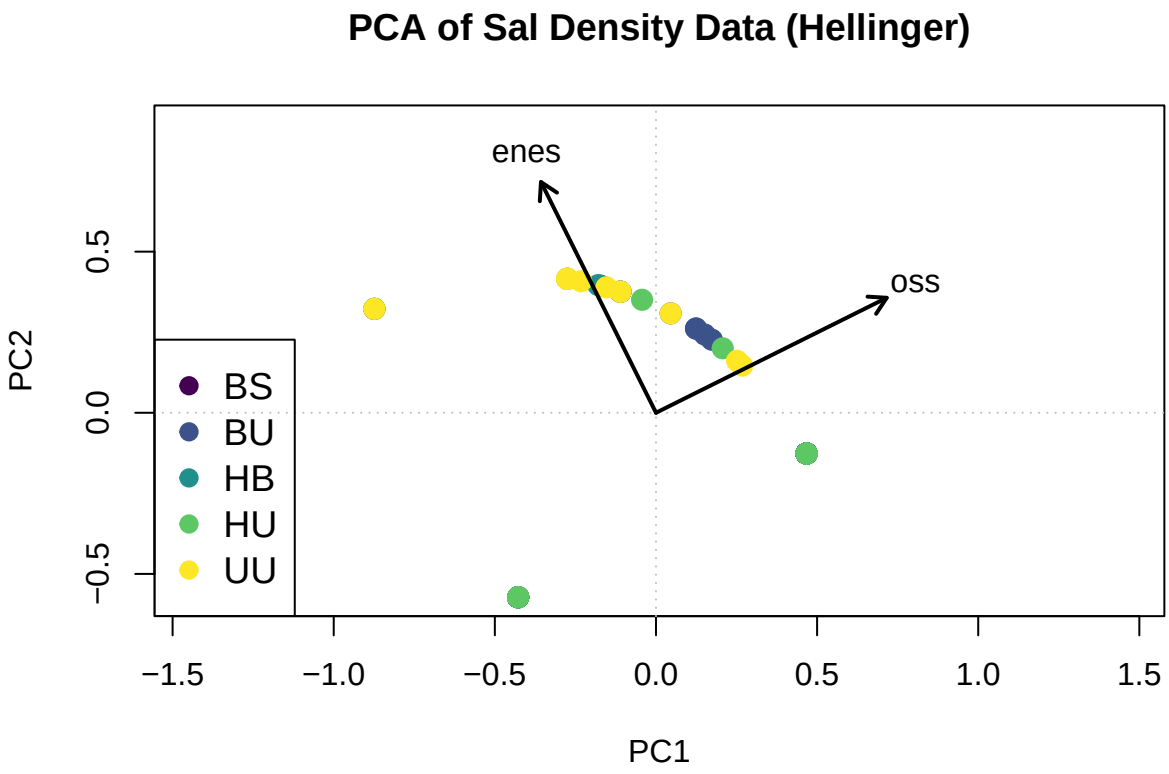
PCA - Sal data

This looks absolutely absurd, I would assume its mainly because I only have two species.

```
## Only 2 axes available
##
## Eigenvectors:

##          PC1    PC2
## oss    0.943 -0.332
## enes   0.332  0.943
```



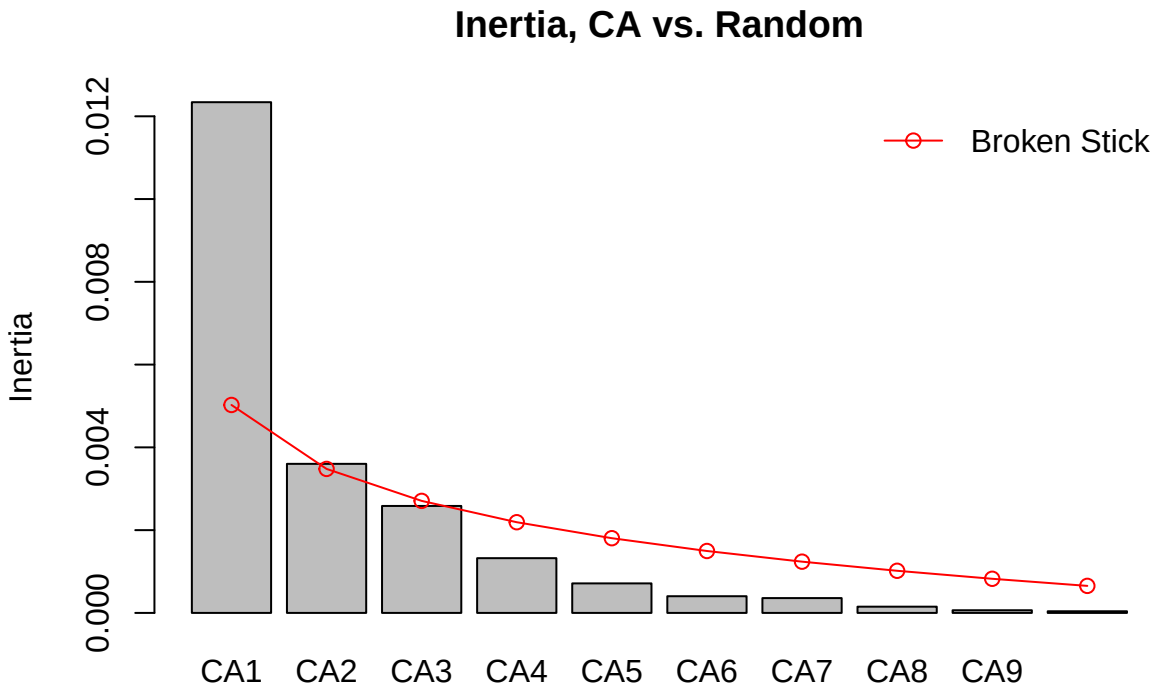

```
## Only 2 axes available
##
## Eigenvectors:

##          PC1    PC2
## oss    0.895  0.446
## enes -0.446  0.895
```

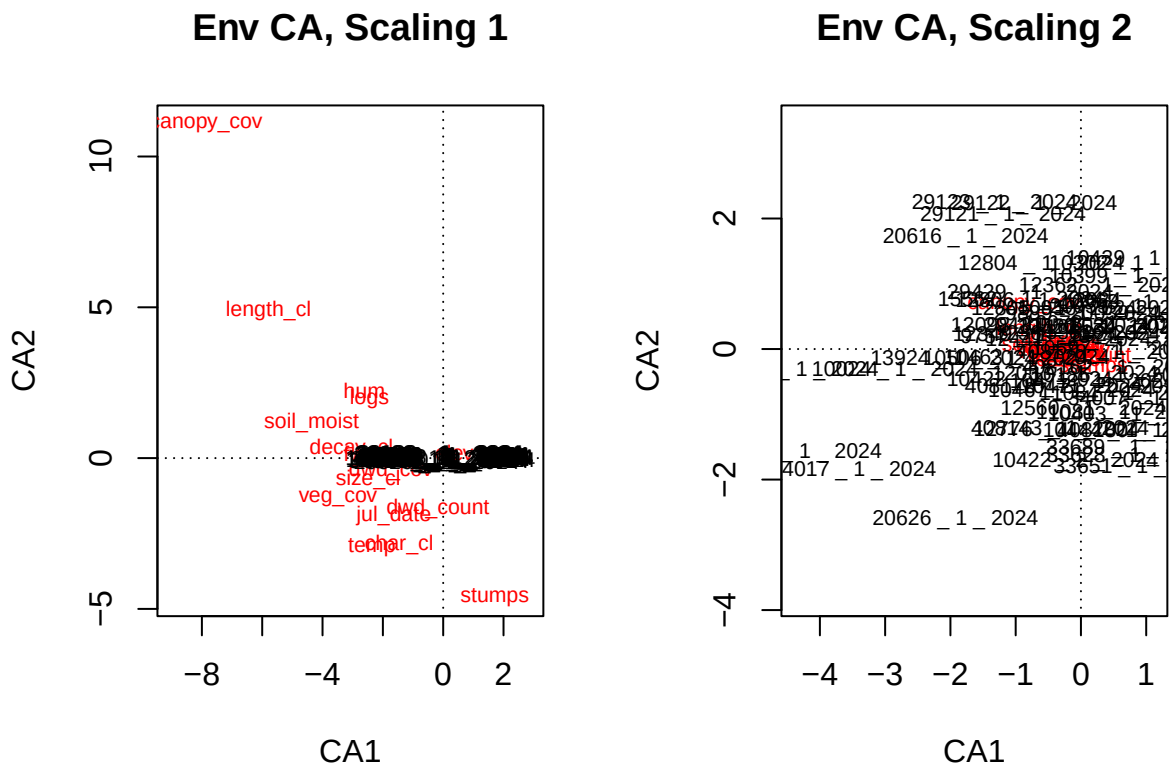


CA - Env data

```
##
## Call:
## cca(X = env_cont)
##
## Partitioning of scaled Chi-square:
##               Inertia Proportion
## Total         0.02164          1
## Unconstrained 0.02164          1
##
## Eigenvalues, and their contribution to the scaled Chi-square
##
## Importance of components:
##                             CA1      CA2      CA3      CA4       CA5       CA6
## Eigenvalue            0.01234 0.003602 0.002585 0.001323 0.0007134 0.0004036
## Proportion Explained  0.57039 0.166478 0.119458 0.061140 0.0329730 0.0186530
## Cumulative Proportion 0.57039 0.736865 0.856323 0.917463 0.9504363 0.9690894
##                             CA7       CA8       CA9      CA10      CA11
## Eigenvalue            0.0003582 0.0001505 6.427e-05 3.995e-05 2.533e-05
## Proportion Explained  0.0165561 0.0069542 2.971e-03 1.846e-03 1.171e-03
## Cumulative Proportion 0.9856454 0.9925996 9.956e-01 9.974e-01 9.986e-01
##                            CA12      CA13      CA14
## Eigenvalue            1.428e-05 1.094e-05 5.335e-06
## Proportion Explained  6.601e-04 5.059e-04 2.466e-04
```

```
## Cumulative Proportion 9.992e-01 9.998e-01 1.000e+00

##       CA1      CA2      CA3      CA4      CA5      CA6      CA7      CA8
## 57.03874 16.64779 11.94578  6.11402  3.29730  1.86530  1.65561  0.69542
##       CA9      CA10     CA11     CA12     CA13     CA14
##   0.29708   0.18464  0.11707  0.06601  0.05059  0.02466
```






```
## null device
##           1
```

CA - Sal data

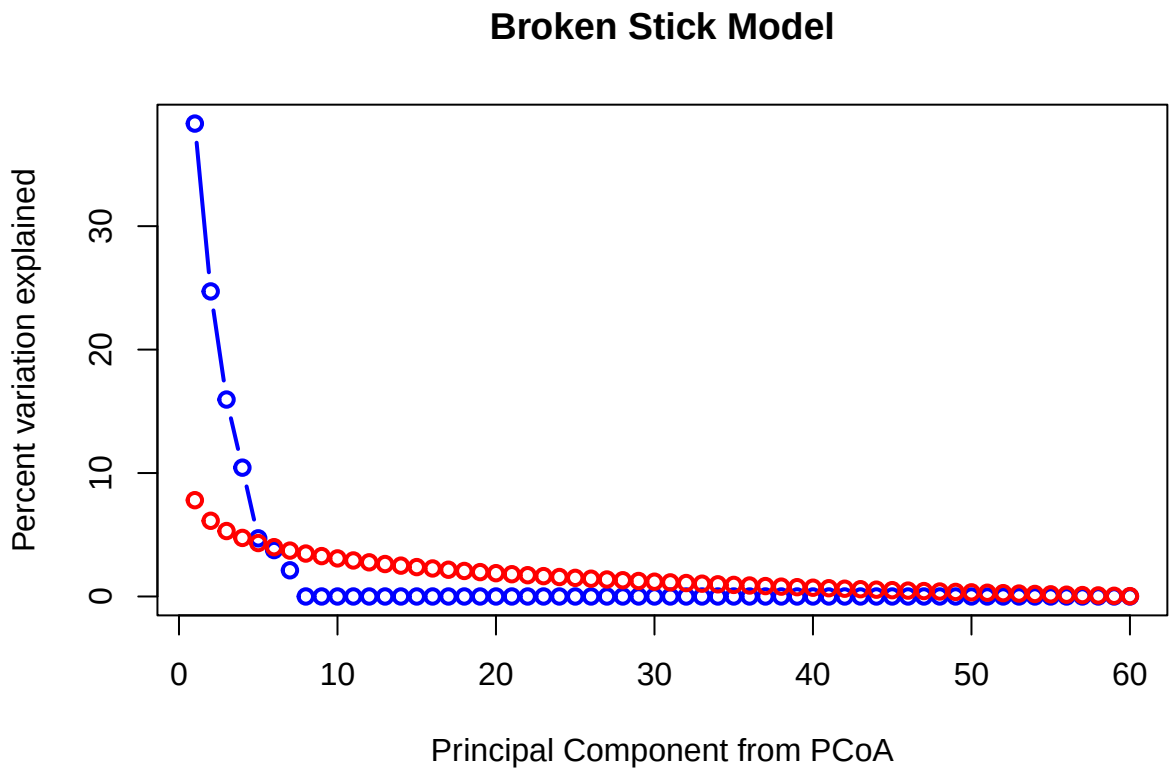
I believe this warning is happening because I have a lot of zeros in my salamander data, but I can't figure out what to do about that. I think this just isnt a great analysis for my data.

```
#sal.ca <- cca(sals)
#Error in cca.default(sals) :
#  all row sums must be >0 in the community data matrix
```

PCoA - Env data

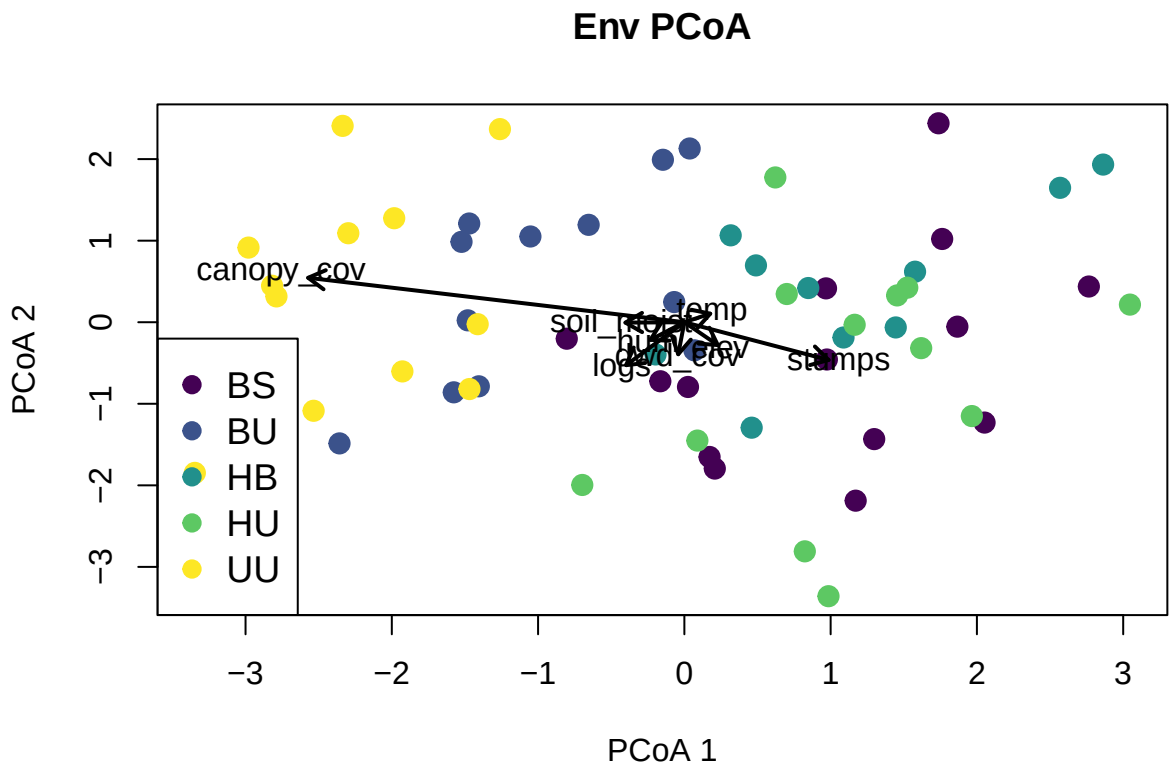
```
#create distance matrix from z-scored subset of env data
        env_std_subset <- env_std[,c("temp","dwd_cov","soil_moist","stumps","logs","decay_cl","canopy_c
        env_euc <- vegdist(env_std_subset, method="euclidean")

        env.pcoa <- cmdscale(env_euc,
                             k=5,
                             eig=TRUE,
                             add=T)
```

```r
#broken stick
        plot(100*env.pcoa$eig/sum(env.pcoa$eig),
            type="b",
            lwd=2,
            col="blue",
            xlab="Principal Component from PCoA",
            ylab="Percent variation explained",
            main="Broken Stick Model")
        lines(bstick(length(env.pcoa$eig))*100,
            type="b",
            lwd=2,
            col="red")
```




```r
        spe.sc <- wascores(env.pcoa$points[,1:2], env_subset)

        vec.sp <- envfit(as.data.frame(env.pcoa$points), env_subset, perm=1000)

#plot pcoa (ordiplot alternative)
        groups <- levels(factor(dat2$trt))
        pt_col <- viridis(length(groups))
        site.sc <- scores(env.pcoa, choices=c(1,2))

plot(site.sc[, 1:2],  # First two dimensions
             main = "Env PCoA",
             xlab = "PCoA 1",
```

```
         ylab = "PCoA 2",
         pch = 19)
for (i in 1:length(groups))
    {
    dim_choice <- site.sc[dat2$trt==groups[i],]
    points(dim_choice[,1], dim_choice[,2],
        pch=19,
        cex=1.4,
        col=pt_col[i])
    }
text(spe.sc*1.5, row.names(spe.sc))
arrows(0, 0, spe.sc[,1]*1.4, spe.sc[,2]*1.4,
    lwd=2,
    length=0.1)
legend(x="bottomleft",
    legend=levels(factor(dat2$trt)),
    col=pt_col[1:6],
    pch=19,
    cex=1.2)
```




```
# Fit environmental variables to PCoA space for interpretation
envfit_results <- envfit(env.pcoa, env_std_subset)
print(envfit_results)
```

```
##
```

```
## ***VECTORS
##
##                Dim1     Dim2     r2 Pr(>r)
## temp        0.73756  0.67528 0.3514  0.001 ***
## dwd_cov    -0.06858 -0.99765 0.7659  0.001 ***
## soil_moist -0.99983 -0.01842 0.3091  0.001 ***
## stumps      0.80849 -0.58851 0.7629  0.001 ***
## logs       -0.43885 -0.89856 0.8121  0.001 ***
## decay_cl   -0.98900  0.14792 0.6418  0.001 ***
## canopy_cov -0.95012  0.31189 0.7685  0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

PCOA - Sal data

Again, I dont think this is useful for my data since I have a lot of zeros.

```
sal.bray <- vegdist(log_sal_dens, "bray")
```

```
## Warning in vegdist(log_sal_dens, "bray"): results may be meaningless because data have negative entri
##                   in method "bray"
```

```
sal.pcoa <- cmdscale(sal.bray,
            k=5,
            eig=TRUE,
            add=T)
```

```
## Warning in cmdscale(sal.bray, k = 5, eig = TRUE, add = T): only 0 of the first
## 5 eigenvalues are > 0
```

NMDS - Env data only

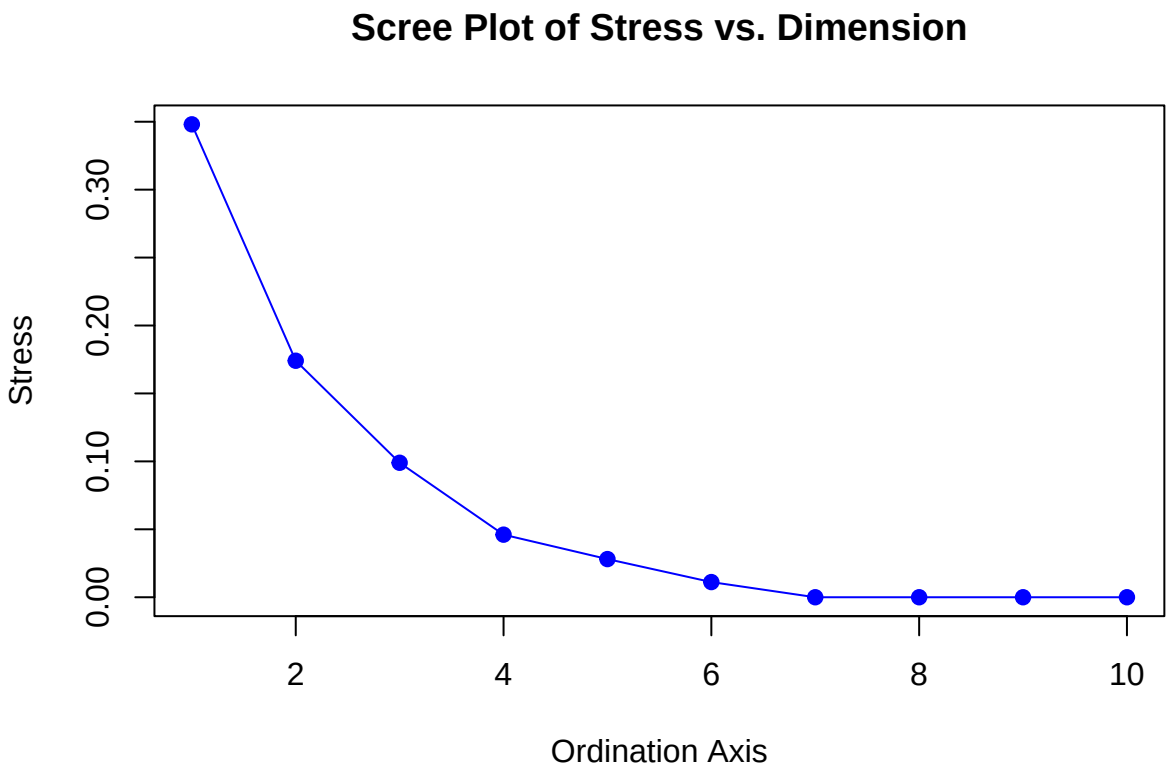
```
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
```

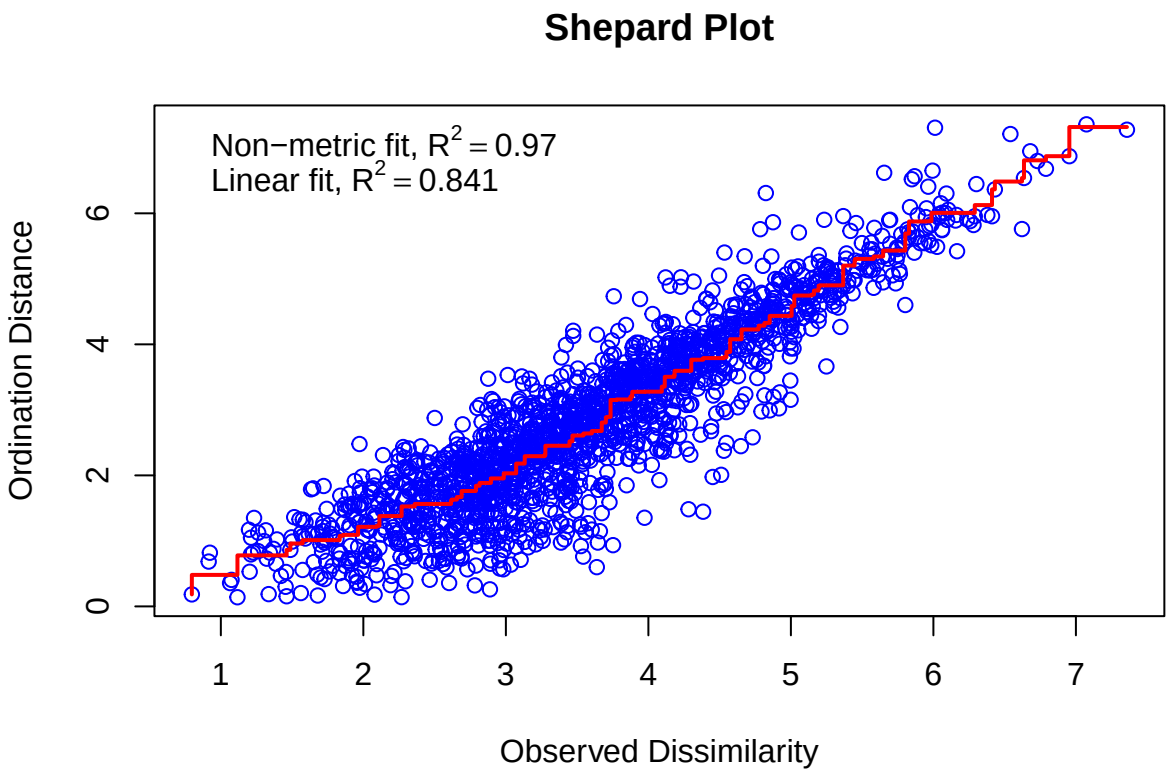
```
## Run 0 stress 0.1740743
## Run 1 stress 0.174237
## ... Procrustes: rmse 0.005032674  max resid 0.02553826
## Run 2 stress 0.1742868
## ... Procrustes: rmse 0.006032315  max resid 0.03043585
## Run 3 stress 0.1740887
## ... Procrustes: rmse 0.001649388  max resid 0.009675061
## ... Similar to previous best
## Run 4 stress 0.1740766
## ... Procrustes: rmse 0.002066923  max resid 0.009792351
## ... Similar to previous best
## Run 5 stress 0.174237
## ... Procrustes: rmse 0.005028381  max resid 0.02553346
## Run 6 stress 0.174237
## ... Procrustes: rmse 0.005021948  max resid 0.02553619
```

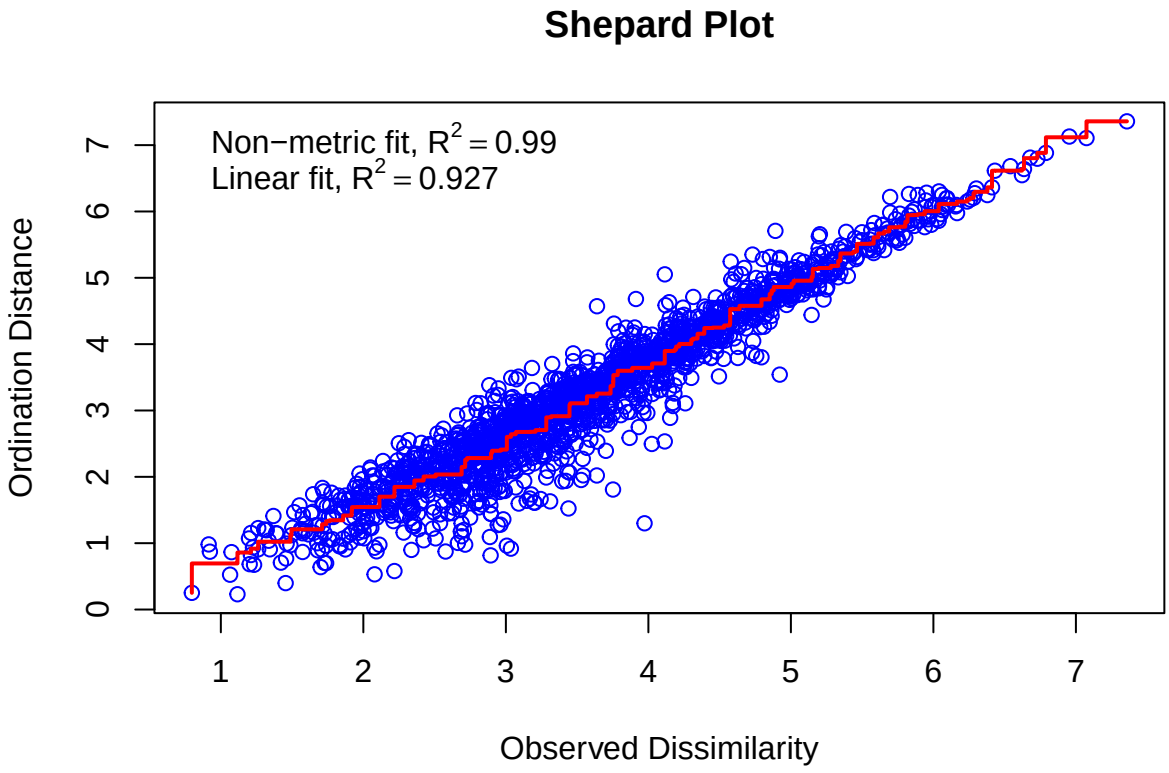
```
## Run 7 stress 0.1742519
## ... Procrustes: rmse 0.005282821  max resid 0.0253712
## Run 8 stress 0.1742511
## ... Procrustes: rmse 0.004875527  max resid 0.02536426
## Run 9 stress 0.1740624
## ... New best solution
## ... Procrustes: rmse 0.00122971  max resid 0.007034305
## ... Similar to previous best
## Run 10 stress 0.1740766
## ... Procrustes: rmse 0.00165056  max resid 0.009695893
## ... Similar to previous best
## Run 11 stress 0.1742519
## ... Procrustes: rmse 0.005162522  max resid 0.02543396
## Run 12 stress 0.1742519
## ... Procrustes: rmse 0.005171929  max resid 0.02543997
## Run 13 stress 0.1740895
## ... Procrustes: rmse 0.002294135  max resid 0.009898852
## ... Similar to previous best
## Run 14 stress 0.1742868
## ... Procrustes: rmse 0.005907962  max resid 0.03022985
## Run 15 stress 0.174237
## ... Procrustes: rmse 0.004912636  max resid 0.02564774
## Run 16 stress 0.174252
## ... Procrustes: rmse 0.005199122  max resid 0.0254382
## Run 17 stress 0.1740766
## ... Procrustes: rmse 0.001652255  max resid 0.009705327
## ... Similar to previous best
## Run 18 stress 0.174237
## ... Procrustes: rmse 0.004907842  max resid 0.02564286
## Run 19 stress 0.1742425
## ... Procrustes: rmse 0.004871095  max resid 0.02530995
## Run 20 stress 0.1740766
## ... Procrustes: rmse 0.001651455  max resid 0.0097003
## ... Similar to previous best
## *** Best solution repeated 5 times

## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

## Run 0 stress 0.09896962
## Run 1 stress 0.09896971
## ... Procrustes: rmse 9.016584e-05  max resid 0.0004376369
## ... Similar to previous best
## Run 2 stress 0.09896972
## ... Procrustes: rmse 8.577676e-05  max resid 0.0004127917
## ... Similar to previous best
## Run 3 stress 0.09896977
## ... Procrustes: rmse 0.0001223393  max resid 0.0005867479
## ... Similar to previous best
## Run 4 stress 0.09896965
## ... Procrustes: rmse 5.866884e-05  max resid 0.0002897183
## ... Similar to previous best
## Run 5 stress 0.09896962
## ... New best solution
## ... Procrustes: rmse 3.259576e-05  max resid 0.0001404313
```

```
## ... Similar to previous best
## Run 6 stress 0.09896965
## ... Procrustes: rmse 8.313361e-05  max resid 0.0003665721
## ... Similar to previous best
## Run 7 stress 0.09896987
## ... Procrustes: rmse 0.000185915  max resid 0.0008367507
## ... Similar to previous best
## Run 8 stress 0.09896993
## ... Procrustes: rmse 0.0001501375  max resid 0.0008206557
## ... Similar to previous best
## Run 9 stress 0.09896962
## ... Procrustes: rmse 2.501235e-05  max resid 0.0001413601
## ... Similar to previous best
## Run 10 stress 0.09896968
## ... Procrustes: rmse 0.0001033617  max resid 0.0004440732
## ... Similar to previous best
## Run 11 stress 0.09896971
## ... Procrustes: rmse 9.430648e-05  max resid 0.0005388081
## ... Similar to previous best
## Run 12 stress 0.09896979
## ... Procrustes: rmse 0.0001556599  max resid 0.000696825
## ... Similar to previous best
## Run 13 stress 0.09896967
## ... Procrustes: rmse 7.68342e-05  max resid 0.0004590056
## ... Similar to previous best
## Run 14 stress 0.09896978
## ... Procrustes: rmse 0.0001239758  max resid 0.0006894128
## ... Similar to previous best
## Run 15 stress 0.09896974
## ... Procrustes: rmse 0.0001364889  max resid 0.0006982714
## ... Similar to previous best
## Run 16 stress 0.09896965
## ... Procrustes: rmse 5.763074e-05  max resid 0.0003457259
## ... Similar to previous best
## Run 17 stress 0.09896967
## ... Procrustes: rmse 7.211913e-05  max resid 0.0004325678
## ... Similar to previous best
## Run 18 stress 0.09896978
## ... Procrustes: rmse 0.0001229242  max resid 0.0007154321
## ... Similar to previous best
## Run 19 stress 0.09896974
## ... Procrustes: rmse 0.0001018879  max resid 0.0005856811
## ... Similar to previous best
## Run 20 stress 0.09896977
## ... Procrustes: rmse 0.0001102717  max resid 0.0006202152
## ... Similar to previous best
## *** Best solution repeated 16 times

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

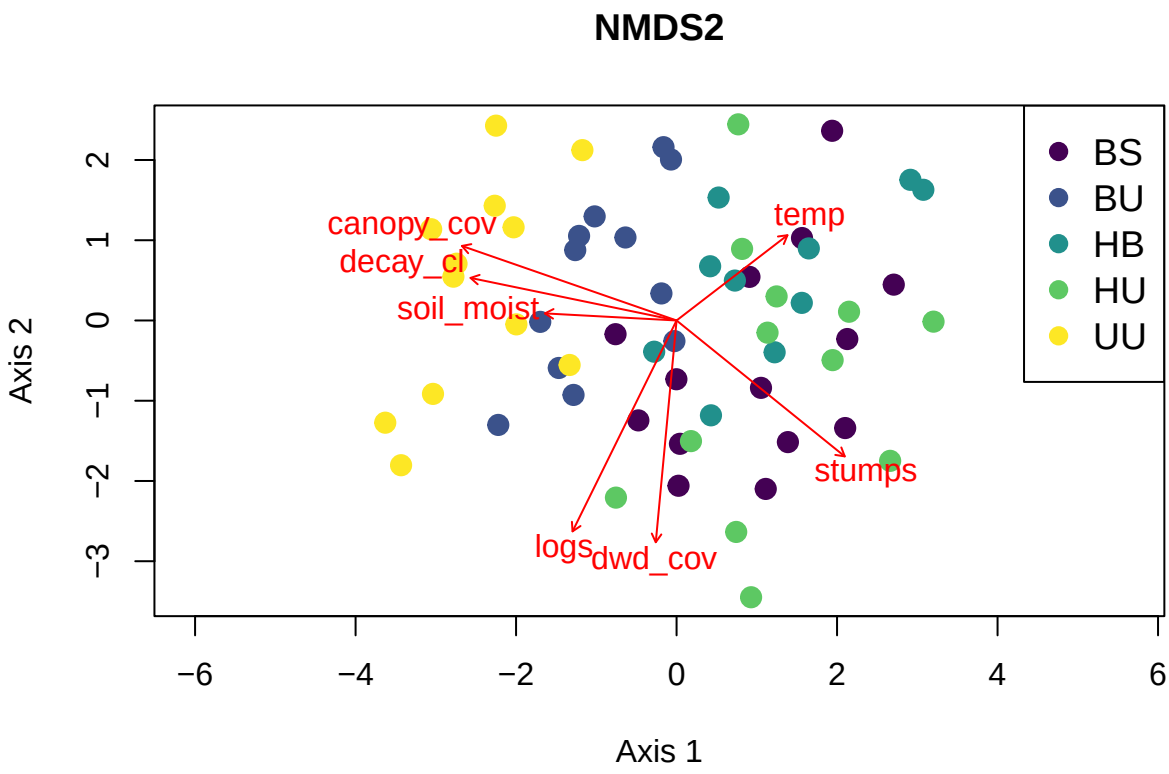
```
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE
## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

## Warning in metaMDS(x, distance = distance, k = i, trymax = trymax,
## autotransform = autotransform, : stress is (nearly) zero: you may have
## insufficient data

## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

## Warning in metaMDS(x, distance = distance, k = i, trymax = trymax,
## autotransform = autotransform, : stress is (nearly) zero: you may have
## insufficient data

## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

## Warning in metaMDS(x, distance = distance, k = i, trymax = trymax,
## autotransform = autotransform, : stress is (nearly) zero: you may have
## insufficient data

## 'comm' has negative data: 'autotransform', 'noshare' and 'wascores' set to FALSE

## Warning in metaMDS(x, distance = distance, k = i, trymax = trymax,
## autotransform = autotransform, : stress is (nearly) zero: you may have
## insufficient data
```


Scree Plot of Stress vs. Dimension
Stress
0.00
0.10
0.20
0.30
2
4
6
8
10
Ordination Axis


Shepard Plot
Non−metric fit, R2 = 0.97
Linear fit, R2 = 0.841
Ordination Distance
Observed Dissimilarity
0
2
4
6
1
2
3
4
5
6
7




```
##
## ***VECTORS
##
##                Dim1     Dim2     r2 Pr(>r)
## temp        0.73756  0.67528 0.3514  0.001 ***
## dwd_cov    -0.06858 -0.99765 0.7659  0.001 ***
## soil_moist -0.99983 -0.01842 0.3091  0.001 ***
## stumps      0.80849 -0.58851 0.7629  0.001 ***
## logs       -0.43885 -0.89856 0.8121  0.001 ***
## decay_cl   -0.98900  0.14792 0.6418  0.001 ***
## canopy_cov -0.95012  0.31189 0.7685  0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999

## species scores not available
```



1a. Were you able to successfuly apply each of these analyses to your dataset? If not, explain why.

Most of these worked with my environmental data, but I dont think I can do species ordinations. I have many zeros, and only two species.

1b. Which transformation or standardization procedures, if any, did you use prior to each analysis?

PCA: z-scored environmental data, hellinger salamander data
CA: raw environmental data
PCoA: euclidean distance matrix from z-scored subset of env data
NMDS: z-scored environmental data

1c. Did you remove any outliers prior to running the ordination procedure? Are there any apparent

outliers in the biplot(s) that appear to warrant further examination?

No, I dont think so.

1d. Which ordination procedure is the most appropriate for your dataset? Defend your answer based on what you know about the strengths and limitations of each procedure.

I think the PCoA is the best option. None of them are showing very clear clusters, but this one does show a grouping of the control sites aside from the rest.

Since euclidean distance is appropriate for my data, I wouldve expected PCA to be more helpful. I alsoexpected the PCoA using euclidean distance to be similar to the PCA, but it was not.

1e. How does the choice of ordination method influence the interpretation of your data?

Different ordination methods can give varying interpretations of your data and need to match your data structure. They can focus on different aspects of the data- for example, CA is more sensitive to rare species. Choosing one that shows absolute (PCA, CA) or relative (NMDS) distances impacts your interpretation as well.

Question 2) For the PCA, PCoA, and/or CA, interpret the meaning of the principal components in the context of your dataset. What do the first few principal components explain about the variation in your data? How do these components relate to the original variables? Are there any variables that are strongly correlated with the principal components?

My PCoA plot of environmental data shows a high loading of canopy cover on axis 1, which may explain the grouping of UU (control) plots on the left side of the plot because they are all unlogged. Im having a harder time understanding the results because apparently every variable is showing as having a significant relationship with each vector. Downed wood is strongly correlated with axis 2, but it doesnt look that way on the plot, so that's odd. Still some workshopping to do.

Question 3) Do a little digging and find at least one creative way of presenting your ordination output that we did not discuss in lab or in class. How does this approach allow you to interpret your data more effectively?

Struggling to make a plot that shows a gradient of downed wood density, so that's in the works. Here's one that shows salamander count by point size. It is interesting to see that the sites in the top right have lower counts than on the left side of the plot.

```
oss_scale <- 1 + log(dat2$oss + 1)
plot(site.sc[, 1:2],  # First two dimensions
     main = "Env PCoA, points sized to OSS count",
     xlab = "PCoA 1",
     ylab = "PCoA 2",
     pch = 19)
for (i in 1:length(groups))
{
    dim_choice <- site.sc[dat2$trt==groups[i],]

}
text(spe.sc*1.5, row.names(spe.sc))
points(site.sc,
       pch=20,
       cex=oss_scale,
       col="purple")
arrows(0, 0, spe.sc[,1]*1.4, spe.sc[,2]*1.4,
```

```
        lwd=2,
        length=0.1)
```