

# Multivariate Modeling Homework

Caroline Tribble

For this assignment will be analyzing data on the Vegetation and Environment in Dutch Dune Meadows.

To import the data and read the metadata run the following:

```
library(vegan)
```

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

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

```
## This is vegan 2.5-6
```

```
data(dune)  
data(dune.env)  
?dune
```

Conduct an indirect ordination on the dune plant community. Specifically, visually examine a NMDS plot using the bray-curtis distance metric. Below is some code to help you develop a potential plot that emphasizes the role of the environmental variable “Moisture”. Describe how you interpret the graphic. What is the goal of creating such a plot? Does this analysis suggest any interesting findings with respect to the dune vegetation?

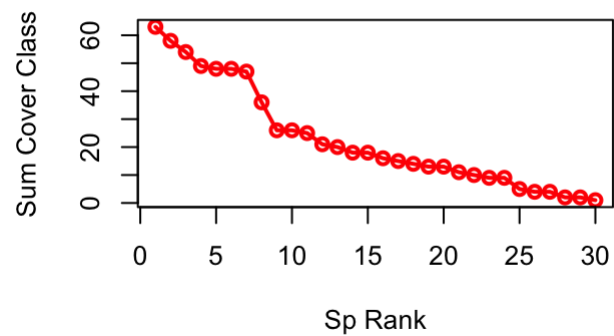
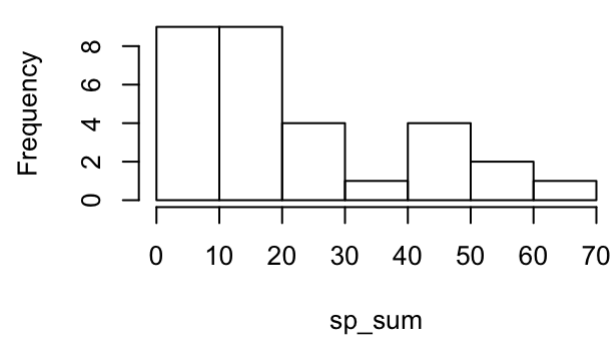
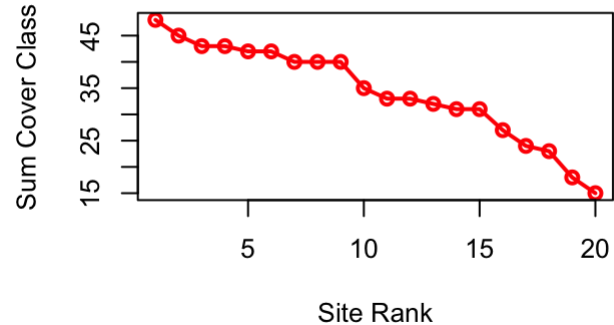
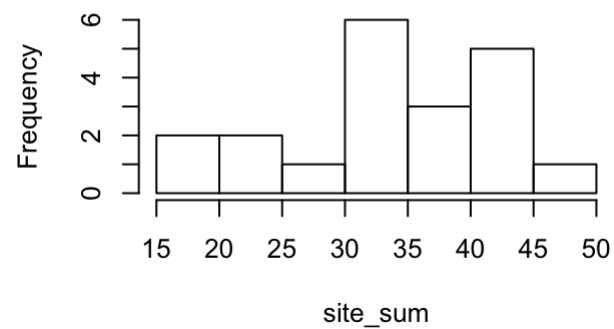
```
str(dune)
```

```
## 'data.frame': 20 obs. of 30 variables:
## $ Achimill: num 1 3 0 0 2 2 2 0 0 4 ...
## $ Agrostol: num 0 0 4 8 0 0 0 4 3 0 ...
## $ Airaprae: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Alop geni: num 0 2 7 2 0 0 0 5 3 0 ...
## $ Anthodor: num 0 0 0 0 4 3 2 0 0 4 ...
## $ Bellpere: num 0 3 2 2 2 0 0 0 0 2 ...
## $ Bromhord: num 0 4 0 3 2 0 2 0 0 4 ...
## $ Chenalbu: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Cirsarve: num 0 0 0 2 0 0 0 0 0 0 ...
## $ Comapalu: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Eleopal u: num 0 0 0 0 0 0 0 4 0 0 ...
## $ Elymrepe: num 4 4 4 4 4 0 0 0 6 0 ...
## $ Empenigr: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Hyporadi: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Juncarti: num 0 0 0 0 0 0 0 4 4 0 ...
## $ Juncbufo: num 0 0 0 0 0 0 2 0 4 0 ...
## $ Lolipere: num 7 5 6 5 2 6 6 4 2 6 ...
## $ Planlanc: num 0 0 0 0 5 5 5 0 0 3 ...
## $ Poaprat : num 4 4 5 4 2 3 4 4 4 4 ...
## $ Poatriv : num 2 7 6 5 6 4 5 4 5 4 ...
## $ Ranuflam: num 0 0 0 0 0 0 0 2 0 0 ...
## $ Rumeacet: num 0 0 0 0 5 6 3 0 2 0 ...
## $ Sagiproc: num 0 0 0 5 0 0 0 2 2 0 ...
## $ Salirepe: num 0 0 0 0 0 0 0 0 0 0 ...
## $ Scorautu: num 0 5 2 2 3 3 3 3 2 3 ...
## $ Trifprat: num 0 0 0 0 2 5 2 0 0 0 ...
## $ Trifrepe: num 0 5 2 1 2 5 2 2 3 6 ...
## $ Vicilath: num 0 0 0 0 0 0 0 0 0 1 ...
## $ Bracruta: num 0 0 2 2 2 6 2 2 2 2 ...
## $ Callcusp: num 0 0 0 0 0 0 0 0 0 0 ...
```

```
sapply(dune, class)
```

```
## Achimill Agrostol Airaprae Alop geni Anthodor Bellpere Bromhord
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
## Chenalbu Cirsarve Comapalu Eleopal u Elymrepe Empenigr Hyporadi
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
## Juncarti Juncbufo Lolipere Planlanc Poaprat Poatriv Ranuflam
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
## Rumeacet Sagiproc Salirepe Scorautu Trifprat Trifrepe Vicilath
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
## Bracruta Callcusp
## "numeric" "numeric"
```

```
# visually explore the cover variable between species and sites
uni_sp = unique(colnames(dune))
sp_sum = apply(dune, 2, sum)
site_sum = apply(dune, 1, sum)
par(mfrow=c(2,2))
hist(sp_sum)
col = colorRamp(c('red', 'orange', 'blue'))
sp_cols = col(length(uni_sp))
plot(sp_sum[order(sp_sum, decreasing=T)], type='o', col='red', lwd=2,
      xlab='Sp Rank', ylab='Sum Cover Class')
hist(site_sum)
plot(site_sum[order(site_sum, decreasing=T)], type='o', col='red', lwd=2,
      xlab='Site Rank', ylab='Sum Cover Class')
```

**Histogram of sp\_sum****Histogram of site\_sum**

```
all.equal(rownames(dune), rownames(dune.env))
```

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

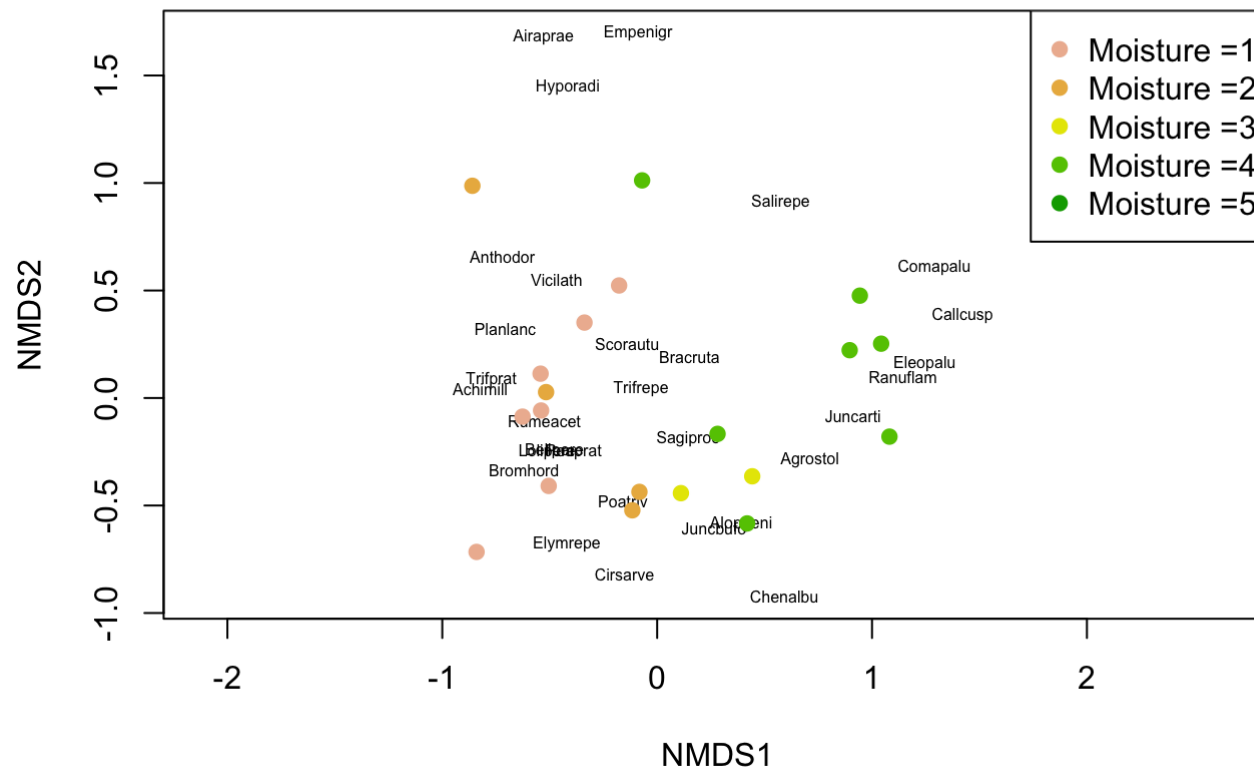
```
dune_mds <- metaMDS(dune)
```

```
## Run 0 stress 0.1192678
## Run 1 stress 0.1183186
## ... New best solution
## ... Procrustes: rmse 0.02026577 max resid 0.06493855
## Run 2 stress 0.1183186
## ... New best solution
## ... Procrustes: rmse 4.047369e-05 max resid 0.0001297714
## ... Similar to previous best
## Run 3 stress 0.1183186
## ... Procrustes: rmse 0.0001300467 max resid 0.000428718
## ... Similar to previous best
## Run 4 stress 0.1183186
## ... Procrustes: rmse 3.371012e-05 max resid 0.0001124658
## ... Similar to previous best
## Run 5 stress 0.1192678
## Run 6 stress 0.119268
## Run 7 stress 0.1183187
## ... Procrustes: rmse 0.000121867 max resid 0.000342935
## ... Similar to previous best
## Run 8 stress 0.1809579
## Run 9 stress 0.1183186
## ... Procrustes: rmse 1.984361e-05 max resid 6.480316e-05
## ... Similar to previous best
## Run 10 stress 0.2035424
## Run 11 stress 0.1192679
## Run 12 stress 0.1192679
## Run 13 stress 0.1808914
## Run 14 stress 0.2035424
## Run 15 stress 0.1192678
## Run 16 stress 0.1192679
## Run 17 stress 0.1192678
## Run 18 stress 0.1192679
## Run 19 stress 0.1889692
## Run 20 stress 0.1183186
## ... Procrustes: rmse 7.4935e-06 max resid 2.537773e-05
## ... Similar to previous best
## *** Solution reached
```

```

plot(dune_mds, type='n')
text(dune_mds, 'sp', cex=.5)
# generate vector of colors
color_vect = rev(terrain.colors(6))[-1]
points(dune_mds, 'sites', pch=19,
       col=color_vect[dune.env$Moisture])
legend('topright', paste("Moisture =", 1:5, sep=''),
      col=color_vect, pch=19)

```



From the NMDS plot we can see how

these different species of vegetation in this dune community relate and differ according to their moisture factor level. Species with high moisture factors (4 and up) are grouped together (including sp. Comapalu, Callcusp, and Eleopal), while those with low moisture factors (2 and below) are grouped together on the opposite side of the graph (including sp. Planlanc, Trifprat, and Achirhill) showing a strong association of moisture level within vegetation species in this community. The NMDS plot is non-eigenbased and based upon Bray-Curtis ecological distance matrices,

meaning only rank information is used and is mapped non-linearly within the ordination space. NMDS plots provide a good un-biased assessment to do in addition to eigenbased since the axes are not ordered and are fitted simultaneously. In the NMDS test run above the plot we can see how it is an iterative process and the stress values represent the goodness of the test. Stress values less than or equal 0.1 are considered fair and values less than or equal to 0.05 support good fit. In this model the stress values range from 0.111 to 0.188 indicating this is a fair ordination of the dune vegetation community.

Carry out a direct ordination using CCA in order to test any potential hypotheses that you developed after examining the MDS plot. Specifically, carry out a test of the entire model (i.e., including all constrained axes) and also carry out tests at the scale of individual explanatory variables you included in your model if you included more than one variable. Plot your results.

```
cca_dune = cca(dune~ ., data=dune.env)
RsquareAdj(cca_dune, 100)
```

```
## $r.squared
## [1] 0.7106267
##
## $adj.r.squared
## [1] 0.2362527
```

Adjusted R squared provides us with model accuracy because as our model complexity increases with the addition of independent variables we often see a skewed increase in our R-squared value. This adjusted R-squared provides a true goodness of fit of the model by taking this impact of model complexity into account. We can see for dune vegetation not all of these explanatory variables need to be included and this is not the best fit.

```
anova(cca_dune, permutations = 999)
```

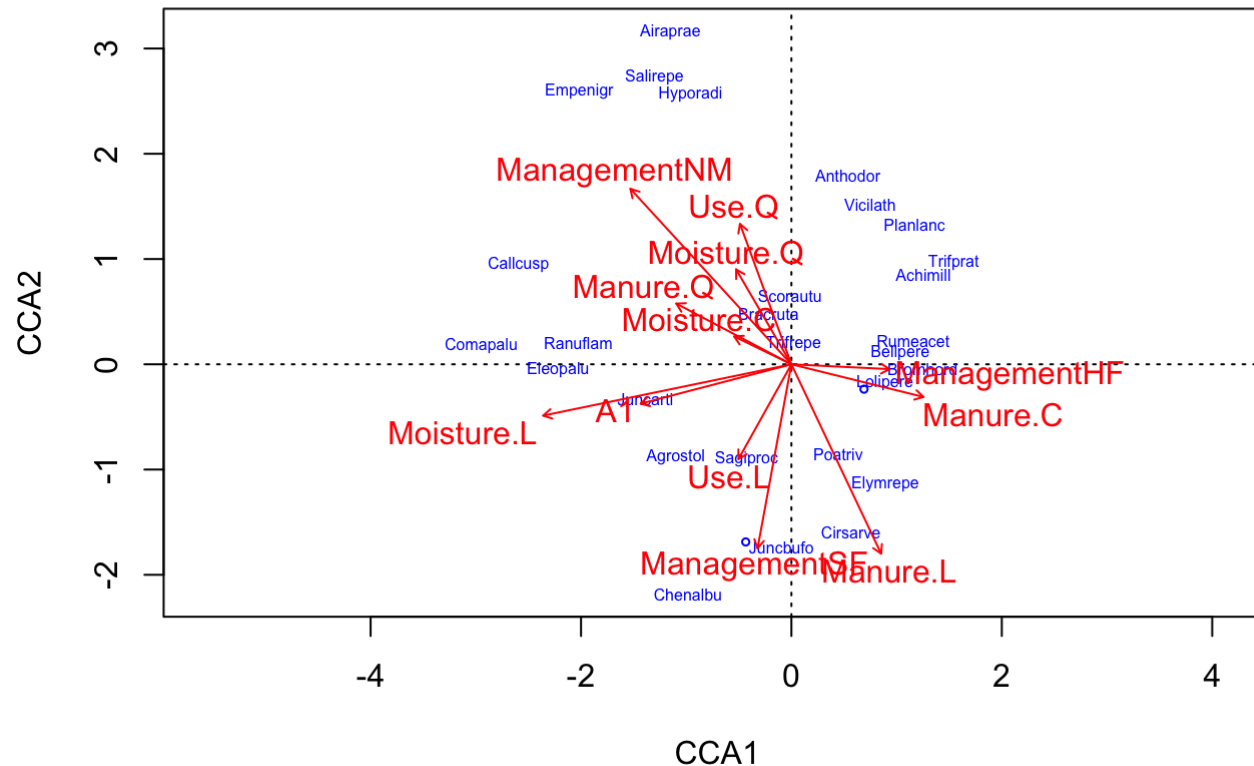
```
## Permutation test for cca under reduced model
## Permutation: free
## Number of permutations: 999
##
## Model: cca(formula = dune ~ A1 + Moisture + Management + Use + Manure, data = dune.env)
##           Df ChiSquare      F Pr(>F)
## Model     12    1.5032 1.4325 0.034 *
## Residual   7     0.6121
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(cca_dune, by='margin', permutations = 999)
```

```
## Permutation test for cca under reduced model
## Marginal effects of terms
## Permutation: free
## Number of permutations: 999
##
## Model: cca(formula = dune ~ A1 + Moisture + Management + Use + Manure, data = dune.env)
##           Df ChiSquare      F Pr(>F)
## A1          1   0.11070 1.2660 0.224
## Moisture     3   0.31587 1.2041 0.250
## Management   2   0.15882 0.9081 0.567
## Use          2   0.13010 0.7439 0.794
## Manure       3   0.25490 0.9717 0.503
## Residual     7   0.61210
```

```
plot(cca_dune, type='n', scaling=1)
orditorp(cca_dune, display='sp', cex=0.5, scaling=1, col='blue')
text(cca_dune, display='bp', col='red')
```





Looking at this graph it is difficult to

determine significance because these are nominal categorical variables and each variable is being broken down by their different nominal groupings within each variable (i.e. Moisture.L, Use.Q, etc.).

CCA similarly to RA and DCA assumes a non-linear species response model and also assumes a unimodal distribution of species along environmental gradients observed. Instead of Bray-Curtis distance matrices, with CCA sample scores are calculated by weighted averaging of species scores and these sample scores are then regressed against the environmental variables. CCA is basically an RA multiple regression that follows the assumptions that species responses to environmental variables are unimodal and that species dispersions are similar across species observed, fitting a gaussian curve. In the plot, species ordination is shown by their position on the xy plane and the length of the arrows for each environmental variable represents the importance of that variable to ordination, while the direction of the arrow indicates the vectors correlation with the xy axes. From this plot it appears Management, Moisture, and Soil Thickness (A1) are the important environmental factors influencing spatial community. ManagementHF is parallel to the positive X axis showing that management factor is highly correlated to that axis. In addition, vectors with angles between two axes (such as ManagementNM) indicate they are correlated with both axes.

```
cca2_dune = cca(dune~Moisture+Management+A1, data=dune.env)
RsquareAdj(cca2_dune, 100)
```

```
## $r.squared
## [1] 0.5385459
##
## $adj.r.squared
## [1] 0.2713351
```

Adjusted R squared actually got worse and it may be because with these nominal categorical variables broken up and not plotted as one variable's effect in one vector it is misleading which environmental variables are most important to include in the model.

```
anova(cca2_dune, permutations = 999)
```

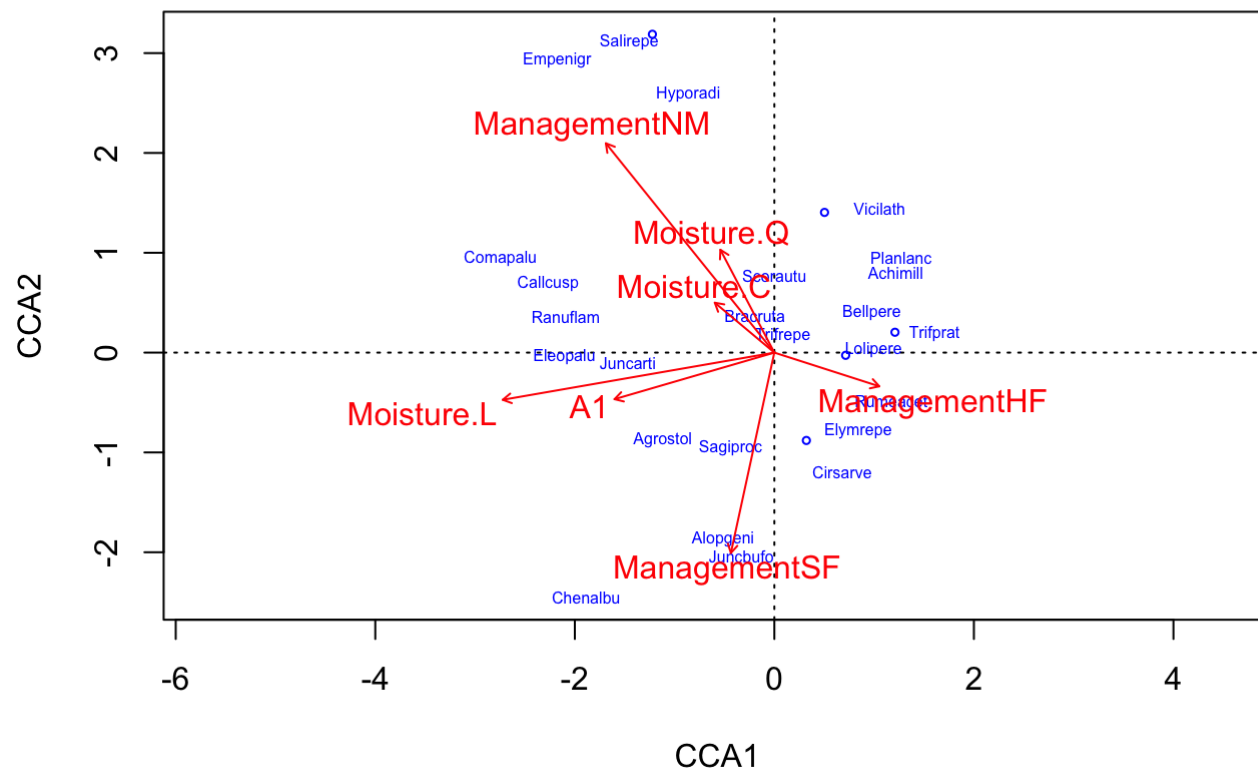
```
## Permutation test for cca under reduced model
## Permutation: free
## Number of permutations: 999
##
## Model: cca(formula = dune ~ Moisture + Management + A1, data = dune.env)
##           Df ChiSquare      F Pr(>F)
## Model      7    1.1392 2.0007 0.001 ***
## Residual 12    0.9761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(cca2_dune, by='margin', permutations = 999)
```

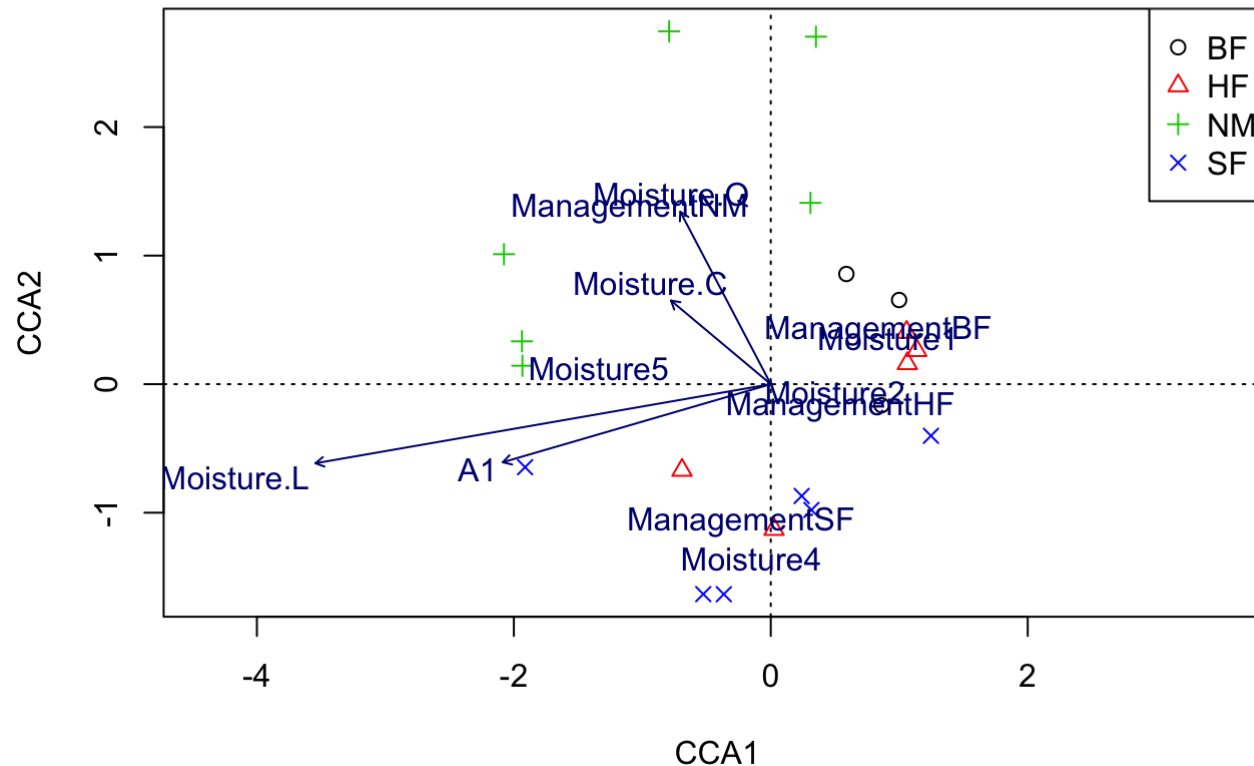
```
## Permutation test for cca under reduced model
## Marginal effects of terms
## Permutation: free
## Number of permutations: 999
##
## Model: cca(formula = dune ~ Moisture + Management + A1, data = dune.env)
##           Df ChiSquare      F Pr(>F)
## Moisture   3   0.35939 1.4728 0.059 .
## Management 3   0.39543 1.6205 0.032 *
## A1         1   0.13679 1.6817 0.081 .
## Residual  12   0.97610
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Management level of the land appears to be the more important factor influencing species distribution in this model followed by moisture and soil thickness.

```
plot(cca2_dune, type='n', scaling=1)
orditorp(cca2_dune, display='sp', cex=0.5, scaling=1, col='blue')
text(cca2_dune, display='bp', col='red')
```



```
type_num = as.numeric(as.factor(dune.env$Management))
ordiplot (cca2_dune, display = c('si', 'cn'), type = 'n')
points (cca2_dune, display = 'si', col = type_num, pch = type_num)
text (cca2_dune, display = 'cn', col = 'navy', cex = 1.0)
legend ('topright', col = 1:4, pch = 1:4, legend = levels (as.factor (dune.env$Management)))
```



This plot focuses on the different

management categories and displays how the distribution of species across management types varies and some management types display more clustering than others.

Do your two analyses agree with one another or complement one another or do these two analyses seem to be suggesting different take home messages? Which analysis do you find to be more useful?

The NMDS plot is an indirect ordination in that the axes plotted are hypothetical whereas the CCA is a direct ordination in that the axes are a linear combination of the environmental variables in the dataset. Since the NMDS plot focuses more on the distribution of species across moisture gradients and the CCA plot shows moisture has a significant effect on species distribution, it has a different take home message in that it shows Management type is the most important factor in understanding the clustering of species. The CCA plot also points to significant trends in soil thickness, which we do not see in the NMDS plot. Usually if a NMDS corroborates a CCA ordination it provides powerful support for this unbiased assessment of the variables done with the NMDS plot. Since we can observe more trends in the CCA plot I would say this analysis is more useful but I am still a bit confused on actually comparing these plots since the NMDS plot focuses on Moisture factor while in the CCA plot we can observe more trends in the dune community data.