I. Combining Mice Data with an extraneous variable: shedding

Download the shedding data text file and scan it in using `scan`

```r
shed = scan("Data/shed.txt")
```

**Exercise 1**

1. Look at this variable, what do you notice? Is it ordinal or nominal? We are going to combine this variable with the previous study we did on these mice, look to see if you have the data `py.nz`, what are its dimensions?

2. How many mice should we use here? Change the data to accommodate for anomalies. (outliers and row sums zero).

```r
shedef = as.factor(shed)[-(70:71)]
pim = py.nz[,-(70:71)]
## We have to deal with the fact that some of the row sums are zero
leaveout = which(apply(pim,1,sum)==0)
pim.nz = pim[-leaveout,]
```

3. Compare the output from the `vegan::cca` function without explanatory variables and the output from `ade4:dudi.coa`.

```r
res.coa = dudi.coa(pim.nz, scannf=FALSE, nf=4)
res.pim = vegan::cca(pim.nz)
```

**Exercise 2**

Using the `vegan` package that you have on the thumbdrive (or have downloaded from the website) use the function `cca` from `vegan` to look at the effect of the shedding variable.

Do a constrained correspondence analysis without any explanatory variables and then with the shedding variable and compare the results and the plots obtained.

```r
res.pim = vegan::cca(pim.nz)
par(mfrow=c(1,2))
plot(res.pim)
title("Output from vegan:cca")
```
Using explanatory covariates  

Exercise 3 1. If you haven’t already, download the data cage96.RData, use this as a factor to project onto the multivariate plot (remember that we take out the two outlying mice), perform a constrained CA using the cagem factor.

```r
> load("Data/cage96.RData")
> cagem=cage96[-(70:71)]
> cagem=factor(cagem)
> res.cage=vegan::cca(t(pim.nz)-cagem)
> plot(res.cage,display="cn")
```
2. Looking at just the effect of shedding
  > res.shed=vegan::cca(t(pim.nz)-shddf)
  > plot(res.shed)
  > text(res.shed, "lc", labels = shddf)
3. In fact we want to condition out the cage variation and look what part of the unconstrained inertia is still explained by the shedf factor. Use the argument perm=1000 to make the permutation test use 1000 simulations

```r
> res.shed.cage=vegan::cca(t(pim.nz)~shedf+Condition(cagem),perm=1000)
```

Permutation test for cca under reduced model

```r
Model: cca(formula = t(pim.nz) ~ shedf + Condition(cagem), perm = 1000)
    Df Chisq F N.Perm Pr(>F)
Model  6 0.1664 1.3271 199 0.01 **
Residual 79 1.6512
---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

4. ```r
> plot(res.shed.cage,display=c("cn"),type="n")
> text(res.shed.cage,display=c("cn"))
```
Exercise 4  What happens if we make shed ordinal?

> shedc=shed[-(70:71)]
> res.shed.cage2=vegan::cca(t(pim.nz)-shedc+Condition(cagem),perm = 1000)
> anova(res.shed.cage2)

Permutation test for cca under reduced model

Model: cca(formula = t(pim.nz) ~ shedc + Condition(cagem), perm = 1000)
 Df  Chisq   F  N.Perm  Pr(>F)
Model   1 0.0305 1.4315    799 0.07875 .
Residual 84 1.7872
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> plot(res.shed.cage2)