

# Two-table methods

in practice

Stéphane Dray

2024-01-08



# Three strategies

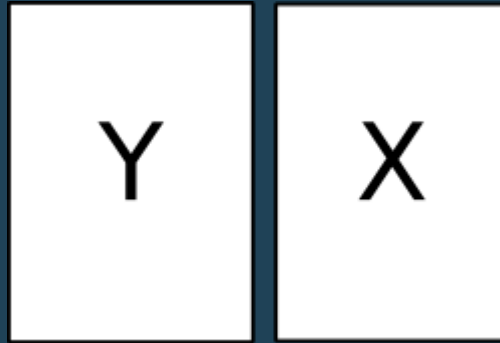


**Canonical Correlation Analysis**

`cancor(tabY, tabX)`

$\text{cor}(\mathbf{Xa}, \mathbf{Yb})$

symmetric



**Co-Inertia Analysis**

`coinertia(dudiY,  
dudiX)`

$\text{cov}(\mathbf{XQa}, \mathbf{YMb})$

symmetric



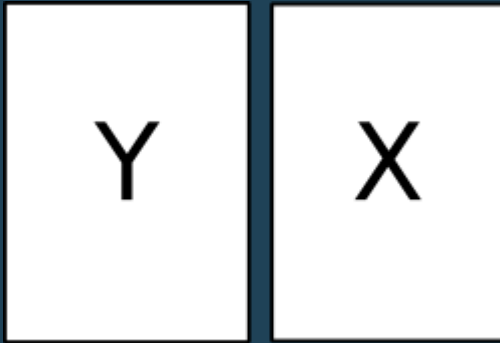
**Instrumental variables**

`pcaiv(dudiY, tabX)`

$\|\widehat{\mathbf{Y}}\mathbf{Qa}\|_{\mathbf{D}}^2$

asymmetric

# Co-inertia analysis

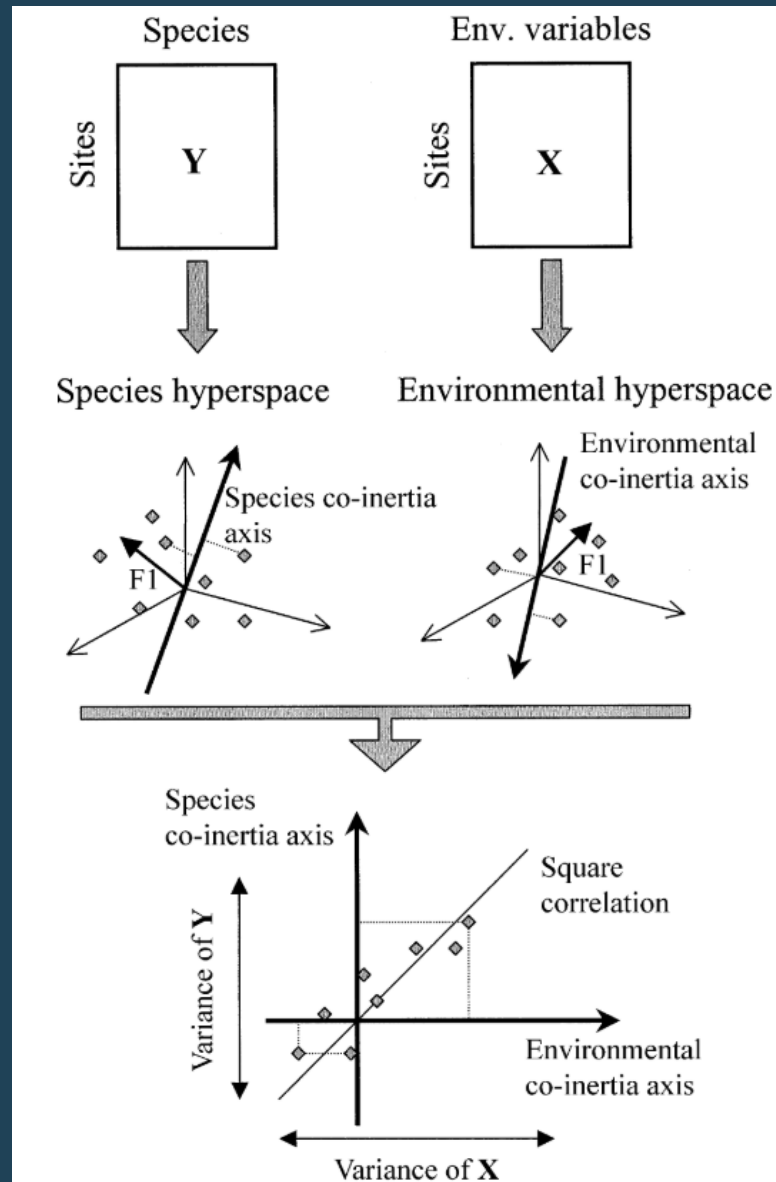


- function `ade4::coinertia`
- maximization of  $cov(\mathbf{XQa}, \mathbf{YMb})$

A compromise between canonical correlation analysis, and the two PCAs:

$$cov(\mathbf{XQa}, \mathbf{YMb}) = cor(\mathbf{XQa}, \mathbf{YMb}) \cdot \|\mathbf{XQa}\|_D \cdot \|\mathbf{YMb}\|_D$$

$$\text{cov}(\mathbf{XQa}, \mathbf{YMb}) = \text{cor}(\mathbf{XQa}, \mathbf{YMb}) \cdot \|\mathbf{XQa}\|_D \cdot \|\mathbf{YMb}\|_D$$

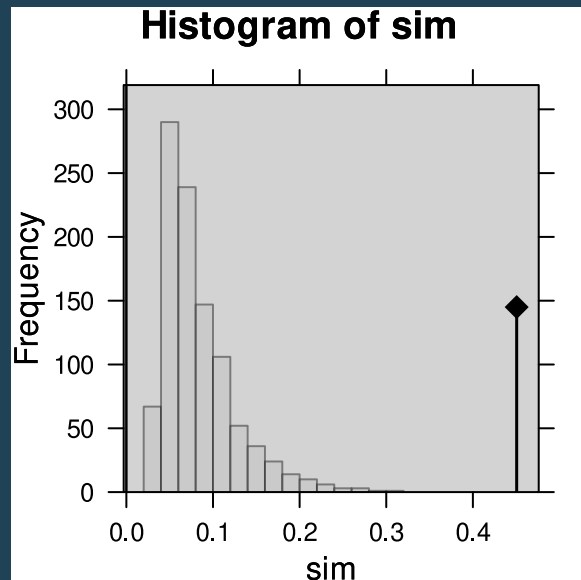


# Perform the analysis

```
library(ade4)
library(adegraphics)
data(doubs)
pca_fish <- dudi.pca(doubs$fish, scale = FALSE, scannf = FALSE)
pca_env <- dudi.pca(doubs$env, scannf = FALSE)
coi <- coinertia(pca_fish, pca_env, scannf = FALSE)
```

# Test the significance of the link

```
rt.coi <- randtest(coi)
plot(rt.coi)
```



```
rt.coi
```

```
## Monte-Carlo test
## Call: randtest.coinertia(xtest = coi)
##
## Observation: 0.4505569
##
## Based on 999 replicates
## Simulated p-value: 0.001
## Alternative hypothesis: greater
##
##          Std.Obs Expectation      Variance
## 8.771899236 0.082839688 0.001757279
```

# Have a look to the summary

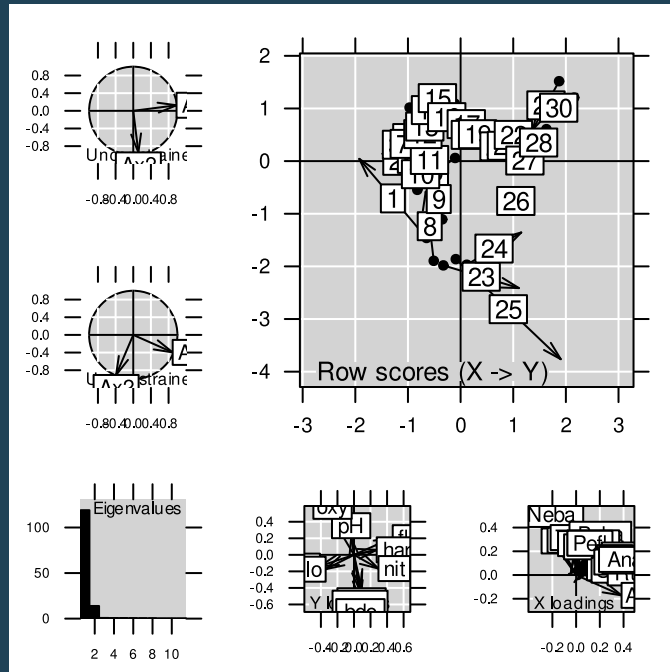
```
summary(coi)
```

```
## Coinertia analysis
##
## Class: coinertia dudi
## Call: coinertia(dudiX = pca_fish, dudiY = pca_env, scannf = FALSE)
##
## Total inertia: 134.7
##
## Eigenvalues:
##      Ax1      Ax2      Ax3      Ax4      Ax5
## 119.0194  13.8714   0.7566   0.5278   0.2709
##
## Projected inertia (%):
##      Ax1      Ax2      Ax3      Ax4      Ax5
## 88.3570 10.2978   0.5617   0.3918   0.2011
##
## ...
```



# Plot the results

```
g1 <- plot(coi)
```

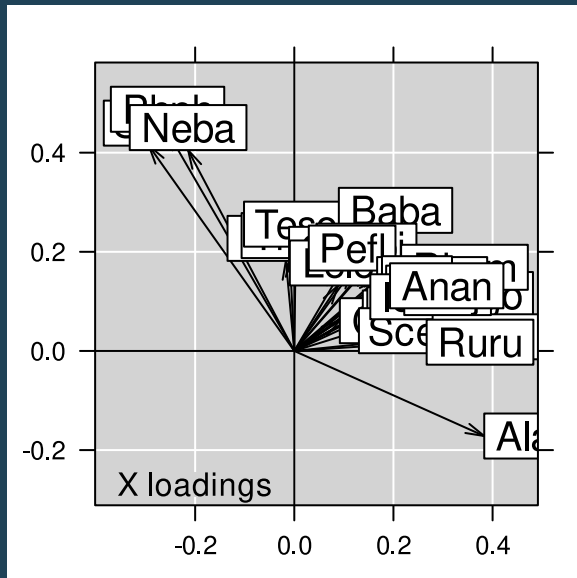


```
names(g1)
```

```
## [1] "Xax"      "Yax"      "eig"      "XYmatch"  "Yloadings"  
## [6] "Xloadings"
```

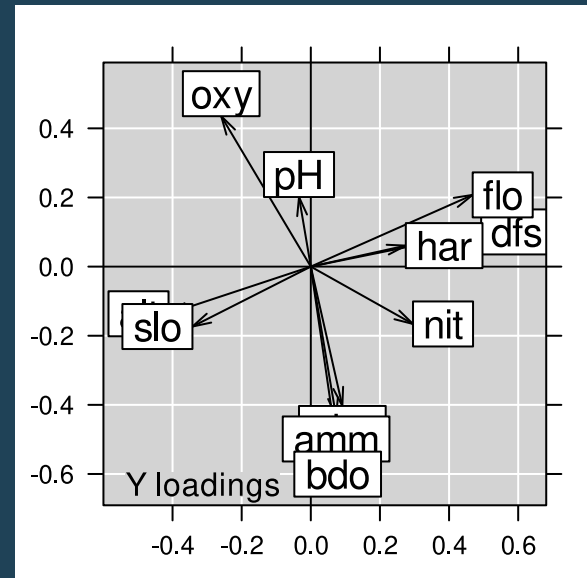
# Loadings for variables

`g1$Xloadings`



**A:** coefficients (loadings) for the variables of **X** (`coi$c1`)

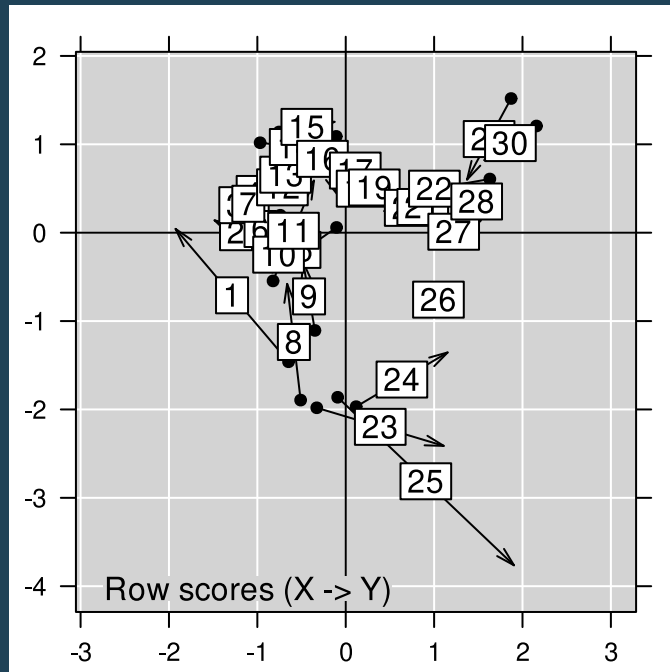
`g1$Yloadings`



**B:** coefficients (loadings) for the variables of **Y** (`coi$l1`)

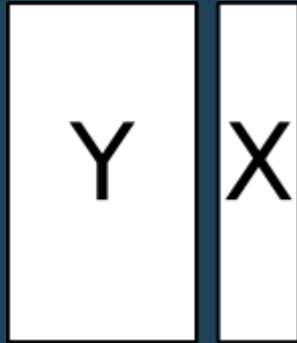
# Scores for individuals

`g1$XYmatch`



- **XQA**: scores of individuals obtained from table **X** (`coi$lX`)
- **YMB**: scores of individuals obtained from table **Y** (`coi$lY`)
- Normed versions (`coi$mX` and `coi$mY`) are represented

# Analysis on instrumental variables



- function `ade4::pcaiv`
- maximization of  $\left\| \widehat{\mathbf{Y}} \mathbf{Q} \mathbf{a} \right\|_{\mathbf{D}}^2$

This framework encompasses:

- Redundancy Analysis (when  $\mathbf{Y}$  is analyzed by principal component analysis by `dudi.pca`)
- Canonical Correspondence Analysis (when  $\mathbf{Y}$  is analyzed by correspondence analysis by `dudi.coa`)

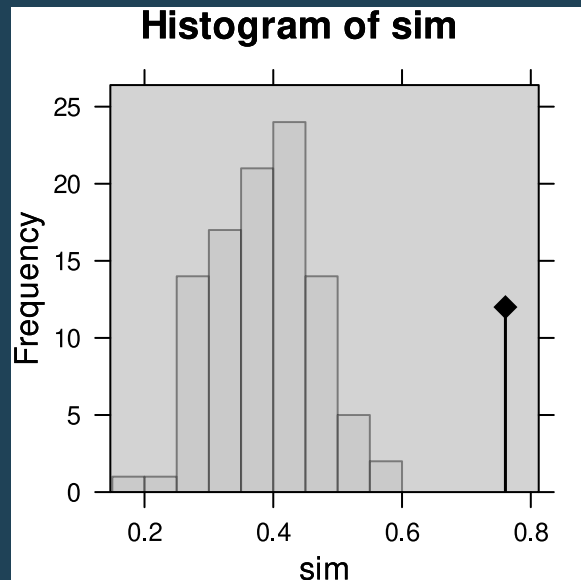
# Perform the analysis

## Redundancy Analysis (RDA)

```
pca_fish <- dudi.pca(doubs$fish, scale = FALSE, scannf = FALSE)  
rda1 <- pcaiv(pca_fish, doubs$env, scannf = FALSE)
```

# Test the significance of the link

```
rt.rda <- randtest(rda1)
plot(rt.rda)
```



```
rt.rda
```

```
## Monte-Carlo test
## Call: randtest.pcaiv(xtest = rda1)
##
## Observation: 0.7605909
##
## Based on 99 replicates
## Simulated p-value: 0.01
## Alternative hypothesis: greater
##
##          Std.Obs Expectation      Variance
## 4.727465961 0.386771952 0.006252673
```

# Have a look to the summary

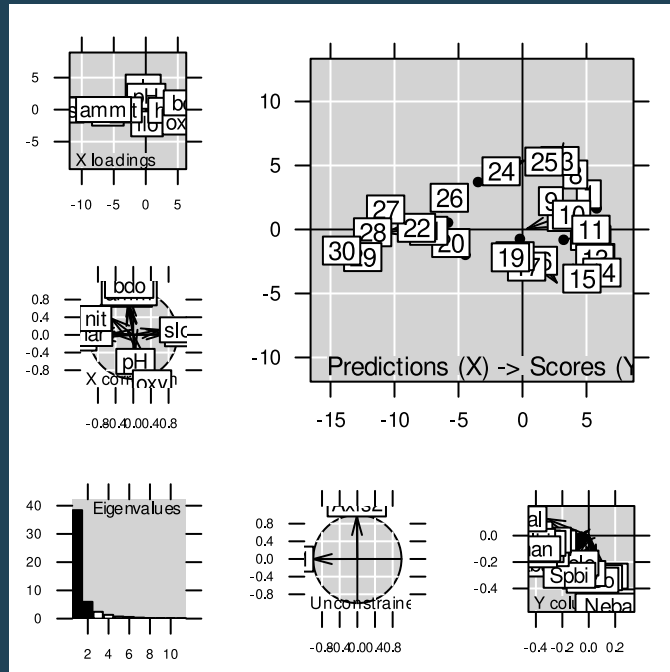
```
summary(rda1)
```

```
## Principal component analysis with instrumental variables
##
## Class: pcaiv dudi
## Call: pcaiv(dudi = pca_fish, df = doubs$env, scannf = FALSE)
##
## Total inertia: 50.26
##
## Eigenvalues:
##      Ax1      Ax2      Ax3      Ax4      Ax5
## 38.4177  5.9540  2.4162  1.3387  0.7431
##
## Projected inertia (%):
##      Ax1      Ax2      Ax3      Ax4      Ax5
##  76.441  11.847   4.808   2.664   1.478
##
...

```

# Plot the results

```
g2 <- plot(rda1)
```



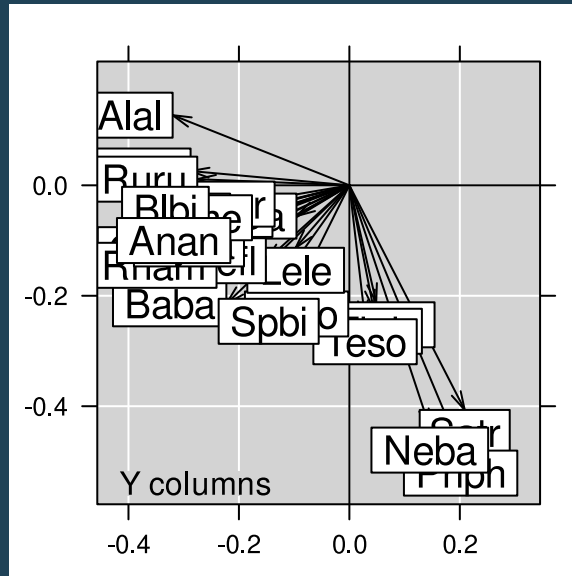
```
names(g2)
```

```
## [1] "Xloadings" "Xcor" "eig" "XYmatch" "Yax"
## [6] "Ycol"
```



# Loadings for variables of Y

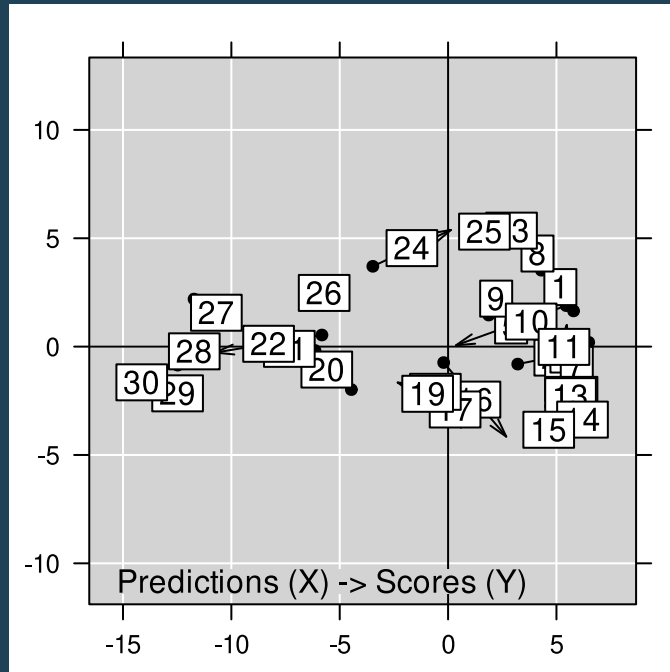
```
g2$Ycol
```



**A**: coefficients (loadings) for the variables of  $\hat{\mathbf{Y}}$  (`rda1$c1`)

# Scores for individuals

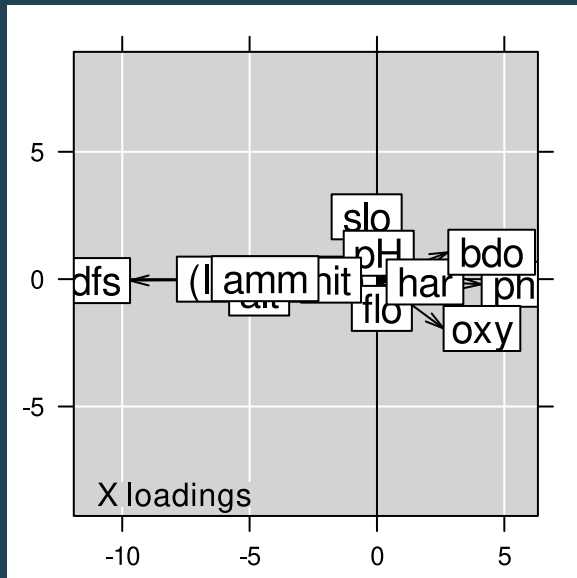
```
g2$XYmatch
```



- $\widehat{\mathbf{Y}}\mathbf{QA}$ : row scores as linear combination of the explanatory variables (`rda1$li`)
- $\mathbf{Y}\mathbf{QA}$ : projection of the rows of  $\mathbf{Y}$  on the principal axes (`rda1$ls`)

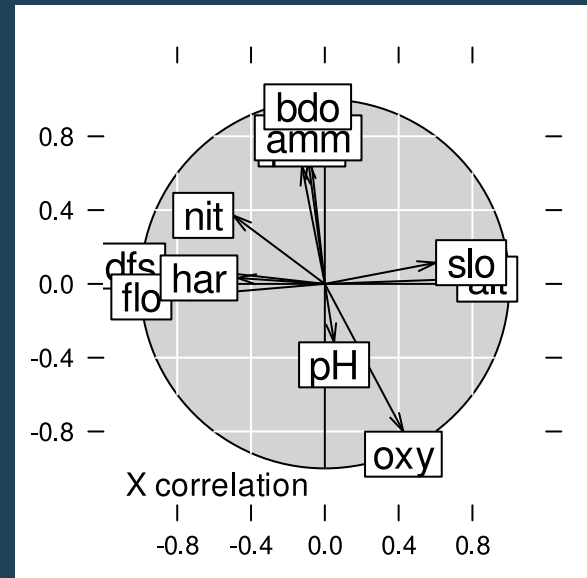
# Variables of X

`g2$Xloadings`



coefficients (loadings) for the variables  
of **X** (`rda1$fa`)

`g2$Xcor`



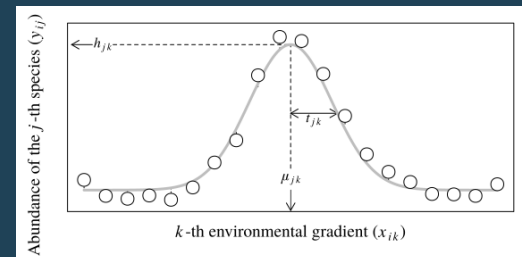
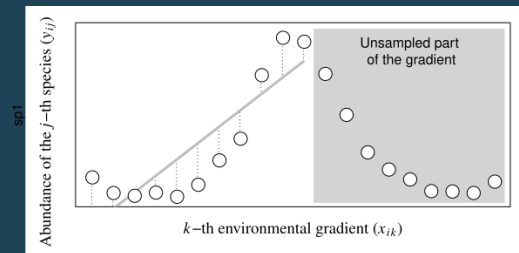
correlation with the variables of **X**  
(`rda1$cor`)

# Your turn

1. Create an Rmd file
2. Load the `rpjdl` data set from `ade4`
3. See `?rpjdl` for details
4. Perform a simple analysis on faunistic data and interpret (either `dudi.pca` or `dudi.coa`)
5. Perform a two-table analysis to relate abundance to environmental data (either `coinertia` or `pcaiv`)
6. Interpret

# PCA or CA on Y?

- PCA assumes a linear response while CA assumes a Gaussian model. Hence, the latter is adapted for long ecological gradients while the former is suitable when only a short part of the gradient is sampled (or to detect decrease in abundances for all species, e.g. effect of pollution)



- In co-inertia analysis, be careful with row weights:

```
coa_fish <- dudi.pca(doubs$fish, scale = FALSE, scannf = FALSE)
pca_env <- dudi.pca(doubs$env, scannf = FALSE, row.w = coa_fish$lw)
coi <- coinertia(coa_fish, pca_env, scannf = FALSE)
```

- Canonical correspondence analysis is simply:

```
cca1 <- pcaiv(coa_fish, doubs$env, scannf = FALSE)
```