Multivariate Analysis

II: Constrained Ordination

Jari Oksanen

Oulu

January 2016

Multivariate Analysis and Ordination

- Basic ordination methods to simplify multivariate data into low dimensional graphics
- Analysis of multivariate dependence and hypotheses
- Analyses can be performed in **R** statistical software using **vegan** package and allies
- Course homepage `http://cc.oulu.fi/~jarioksa/opetus/metodi/`
- **Vegan** homepage `https://github.com/vegandevs/vegan/`

Outline

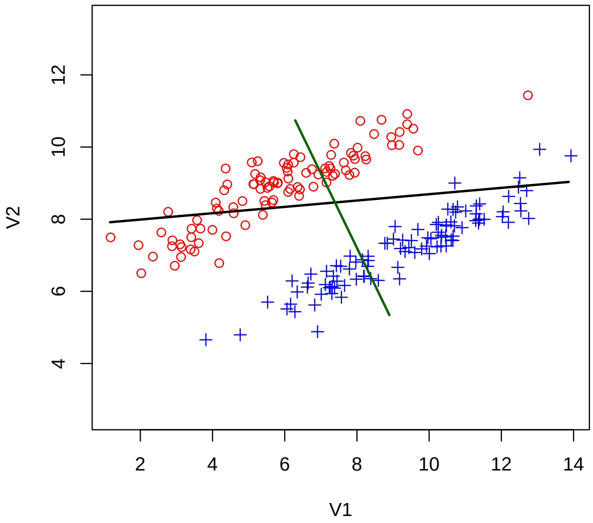
1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis
2. Analysis of Dissimilarities
   - Methods

Outline

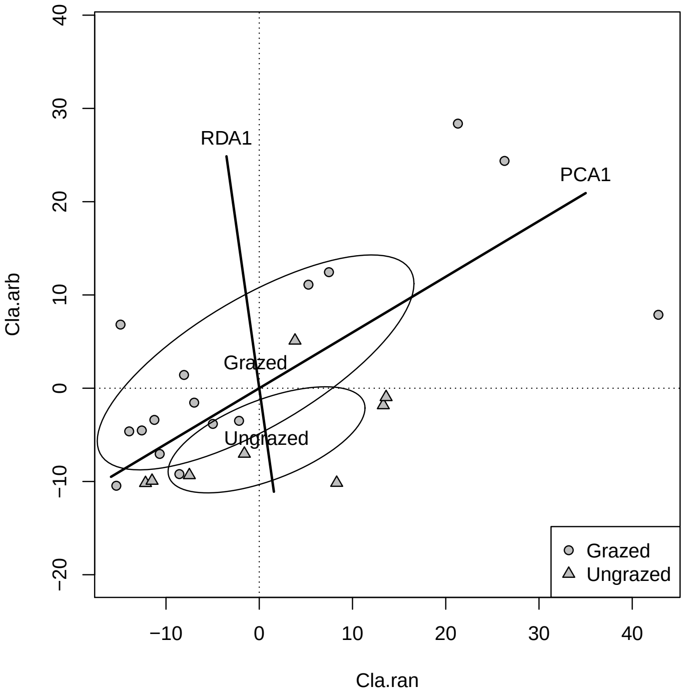
1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis
2. Analysis of Dissimilarities
   - Methods

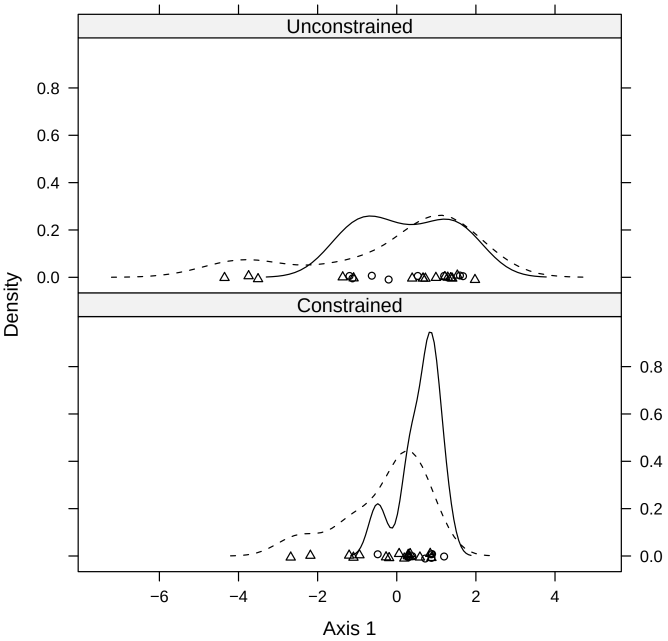
Constrained vs. Unconstrained

- Unconstrained ordination tries to display the variation in data.
- Constrained ordination tries to display only the variation that can be explained with constraining variables.
- You can only observe things that you have measured.



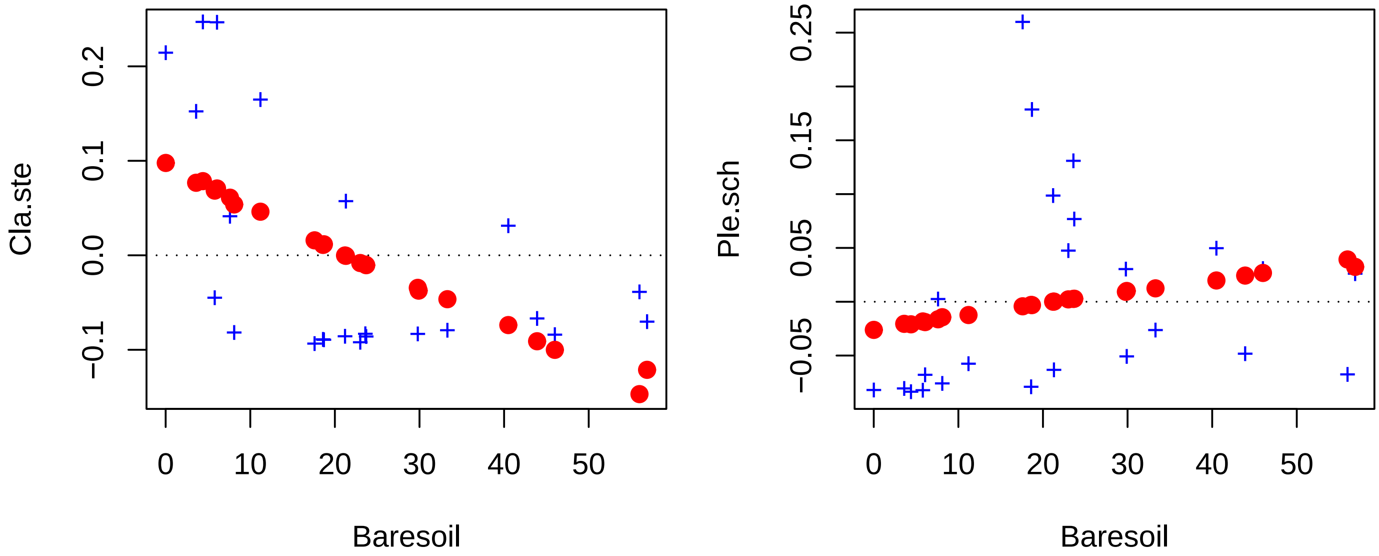
The Idea of Constrained Ordination: Application



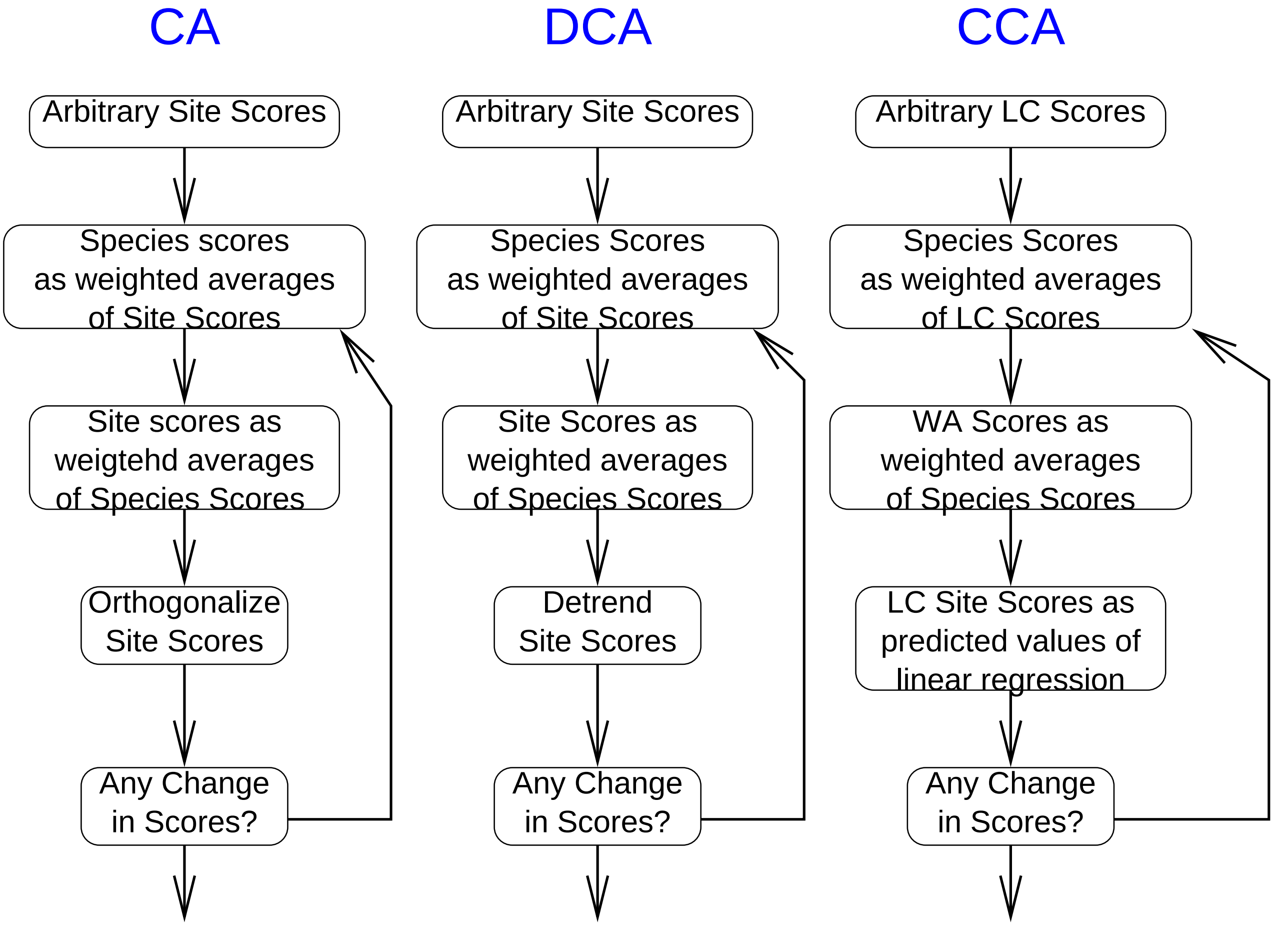


Constrained CA

1. Fit weighted linear regression to all species individually using all constraints as explanatory variables
2. Analyse fitted values using CA



Alternative Algoritm: Alternate Regression and WA



CA

Arbitrary Site Scores

Species scores as weighted averages of Site Scores

Site scores as weigtehd averages of Species Scores

Orthogonalize Site Scores

Any Change in Scores?

DCA

Arbitrary Site Scores

Species Scores as weighted averages of Site Scores

Site Scores as weighted averages of Species Scores

Detrend Site Scores

Any Change in Scores?

CCA

Arbitrary LC Scores

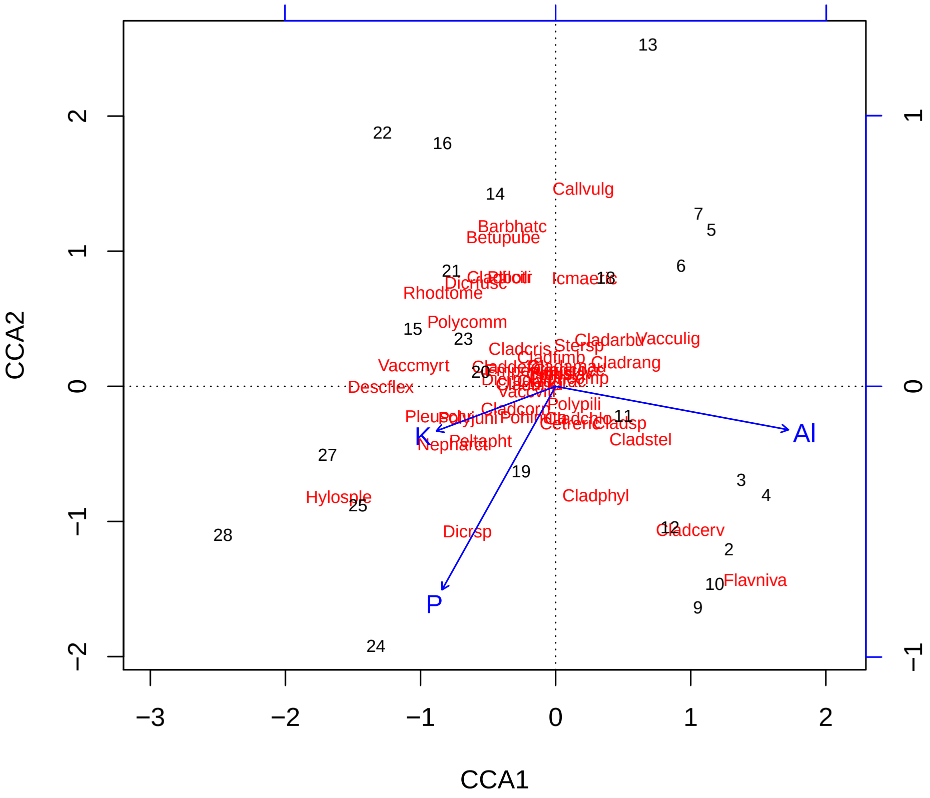
Species Scores as weighted averages of LC Scores

WA Scores as weighted averages of Species Scores

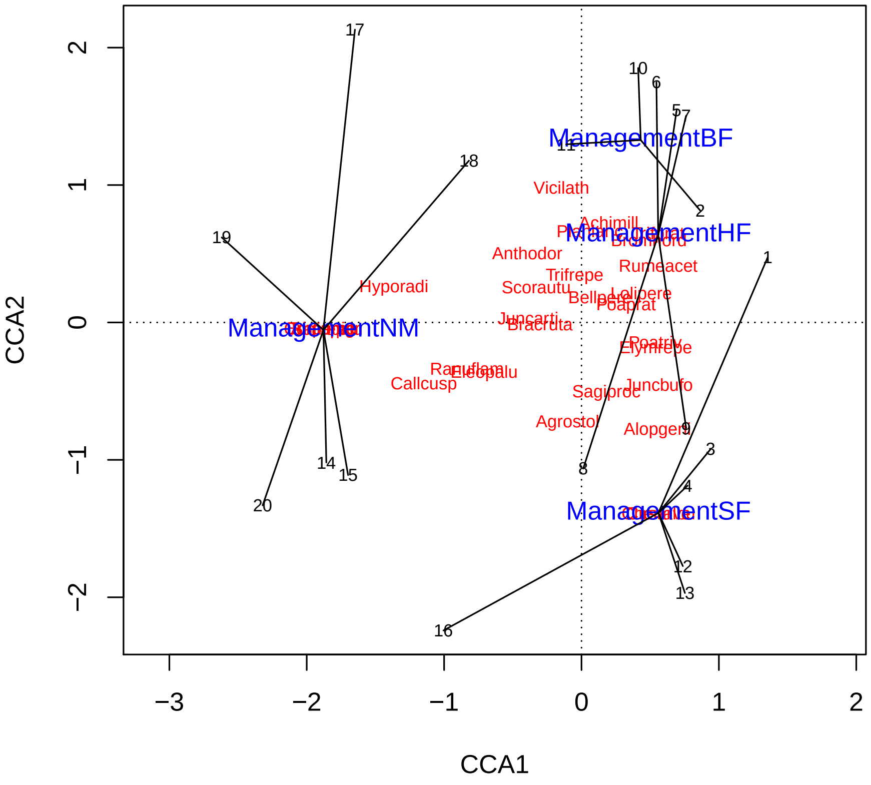
LC Site Scores as predicted values of linear regression

Any Change in Scores?

Example: Continuous Constraints



Example: Class Constraints



Constrained Ordination

1. Distance-based Redundancy Analysis (db-RDA) in function `capscale` is related to metric multidimensional scaling (`cmdscale`). It can handle any dissimilarity measures and performs a linear mapping.
2. Redundancy analysis (RDA) in function `rda` is related to principal components analysis. It is based on Euclidean distances and performs linear mapping.
3. Constrained correspondence analysis (CCA) in function `cca` is related to correspondence analysis. It is based on Chi-squared distances and performs weighted linear mapping.

Running CCA I

```
> (ord <- cca(varespec, varechem))

Call: cca(X = varespec, Y = varechem)

              Inertia Proportion Rank
Total           2.083      1.000
Constrained     1.441      0.692   14
Unconstrained   0.642      0.308    9
Inertia is mean squared contingency coefficient

Eigenvalues for constrained axes:
 CCA1  CCA2  CCA3  CCA4  CCA5  CCA6  CCA7  CCA8  CCA9 CCA10
0.439 0.292 0.163 0.142 0.118 0.089 0.070 0.058 0.031 0.013
CCA11 CCA12 CCA13 CCA14
0.008 0.007 0.006 0.005

Eigenvalues for unconstrained axes:
   CA1    CA2    CA3    CA4    CA5    CA6    CA7    CA8    CA9
0.1978 0.1419 0.1012 0.0708 0.0533 0.0333 0.0189 0.0151 0.0095

> head(summary(ord), 3)
```

Running CCA II

```
Call:
cca(X = varespec, Y = varechem)

Partitioning of mean squared contingency coefficient:
              Inertia Proportion
Total           2.083      1.000
Constrained     1.441      0.692
Unconstrained   0.642      0.308

Eigenvalues, and their contribution to the mean squared contingency coefficient

Importance of components:
                       CCA1  CCA2   CCA3   CCA4   CCA5   CCA6
Eigenvalue            0.439 0.292 0.1628 0.1421 0.1180 0.0890
Proportion Explained  0.211 0.140 0.0782 0.0682 0.0566 0.0427
Cumulative Proportion 0.211 0.351 0.4289 0.4971 0.5537 0.5965
                        CCA7   CCA8   CCA9   CCA10   CCA11
Eigenvalue            0.0703 0.0584 0.0311 0.01329 0.00836
Proportion Explained  0.0337 0.0280 0.0149 0.00638 0.00402
Cumulative Proportion 0.6302 0.6583 0.6732 0.67958 0.68359
                        CCA12   CCA13   CCA14    CA1    CA2
Eigenvalue            0.00654 0.00616 0.00473 0.1978 0.1419
```

Running CCA III

```
Proportion Explained  0.00314 0.00296 0.00227 0.0949 0.0681
Cumulative Proportion 0.68673 0.68969 0.69196 0.7869 0.8550
                         CA3    CA4    CA5    CA6     CA7
Eigenvalue            0.1012 0.0708 0.0533 0.0333 0.01887
Proportion Explained  0.0486 0.0340 0.0256 0.0160 0.00906
Cumulative Proportion 0.9036 0.9376 0.9631 0.9791 0.98820
                          CA8     CA9
Eigenvalue            0.01510 0.00949
Proportion Explained  0.00725 0.00455
Cumulative Proportion 0.99545 1.00000

Accumulated constrained eigenvalues
Importance of components:
                       CCA1  CCA2  CCA3   CCA4   CCA5   CCA6
Eigenvalue            0.439 0.292 0.163 0.1421 0.1180 0.0890
Proportion Explained  0.304 0.202 0.113 0.0986 0.0818 0.0618
Cumulative Proportion 0.304 0.507 0.620 0.7184 0.8003 0.8620
                        CCA7   CCA8   CCA9   CCA10   CCA11
Eigenvalue            0.0703 0.0584 0.0311 0.01329 0.00836
Proportion Explained  0.0488 0.0405 0.0216 0.00922 0.00580
Cumulative Proportion 0.9108 0.9513 0.9729 0.98211 0.98791
                        CCA12   CCA13   CCA14
```

Running CCA IV

```
Eigenvalue           0.00654 0.00616 0.00473
Proportion Explained  0.00454 0.00427 0.00328
Cumulative Proportion 0.99245 0.99672 1.00000

Scaling 2 for species and site scores
* Species are scaled proportional to eigenvalues
* Sites are unscaled: weighted dispersion equal on all dimensions


Species scores

            CCA1    CCA2   CCA3   CCA4   CCA5    CCA6
Callvulg  0.0753 -0.9358 1.6777  0.696  1.078 -0.3450
Empenigr -0.1813  0.0761 0.0365 -0.428 -0.138  0.0105
Rhodtome -1.0535 -0.0603 0.0774 -0.939 -0.214 -0.5180
....


Site scores (weighted averages of species scores)

       CCA1   CCA2   CCA3   CCA4   CCA5  CCA6
18    0.178 -1.060 -0.409 -0.607 -0.565 0.242
```

Running CCA V

```
15   -0.970 -0.197  0.421  0.303  0.152 0.804
24   -1.280  0.476 -2.947  0.393  3.954 0.766
....


Site constraints (linear combinations of constraining variables)

       CCA1   CCA2   CCA3   CCA4    CCA5   CCA6
18   -0.423 -1.325 -0.492 -0.945 -0.0485  0.940
15   -0.190  0.497  0.455 -0.530 -0.0766 -0.790
24   -0.863  0.252 -2.760  0.570  3.2927  0.263
....


Biplot scores for constraining variables

           CCA1    CCA2     CCA3    CCA4     CCA5     CCA6
N        -0.223 -0.5287  0.00685  0.1778 -0.25359  0.10258
P        -0.319  0.5790 -0.16203  0.4795  0.18418 -0.12198
K        -0.366  0.3080  0.35983  0.4795  0.32551 -0.19676
Ca       -0.448  0.4218 -0.03779  0.0982  0.30808  0.04346
Mg       -0.435  0.3407 -0.14216  0.1080  0.49788 -0.00570
```

Running CCA VI

```
S        -0.024  0.4159  0.14840  0.4446  0.59712 -0.16631
Al        0.770 -0.0477  0.03755  0.3909  0.16111 -0.33702
Fe        0.649 -0.0886 -0.04218  0.2627 -0.06955 -0.11188
Mn       -0.722  0.2247  0.11306  0.2916 -0.13870  0.18055
Zn       -0.358  0.3352 -0.27789  0.3460  0.61920 -0.00103
Mo        0.205 -0.1028 -0.15689  0.3250  0.51625 -0.31305
Baresoil -0.537 -0.2538  0.13751 -0.5202  0.16592 -0.35143
Humdepth -0.697  0.2023  0.27184 -0.1353 -0.00363 -0.05074
pH        0.497  0.0744 -0.32666  0.0203 -0.14517 -0.05996
```

Numbers

- Eigenvalues and axis scores like in unconstrained ordination
- Eigenvalues should be lower than in unconstrained analysis, or constraints had no effect
- Components separately for constrained (explained) and unconstrained (residual) variation
- Four kind of scores
  1. Species scores derived from site (LC) scores
  2. Site scores which are linear components of constraints: **LC Scores**
  3. Site scores derived from species scores: **WA Scores**
  4. Scores for constraints: arrowheads for continuous variables (**biplot** scores) and centroids of factor levels
- Species–environment correlation: correlation between WA and LC scores

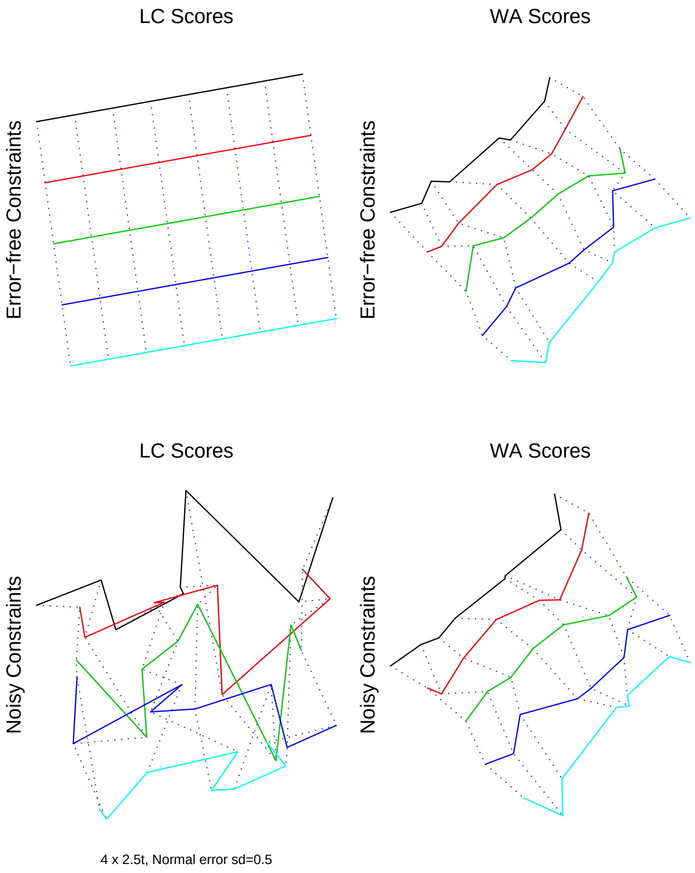
WA or LC Scores?

Mike Palmer:

- Use LC scores, because they give the best fit with the environment, and WA scores are a step from CCA towards CA.

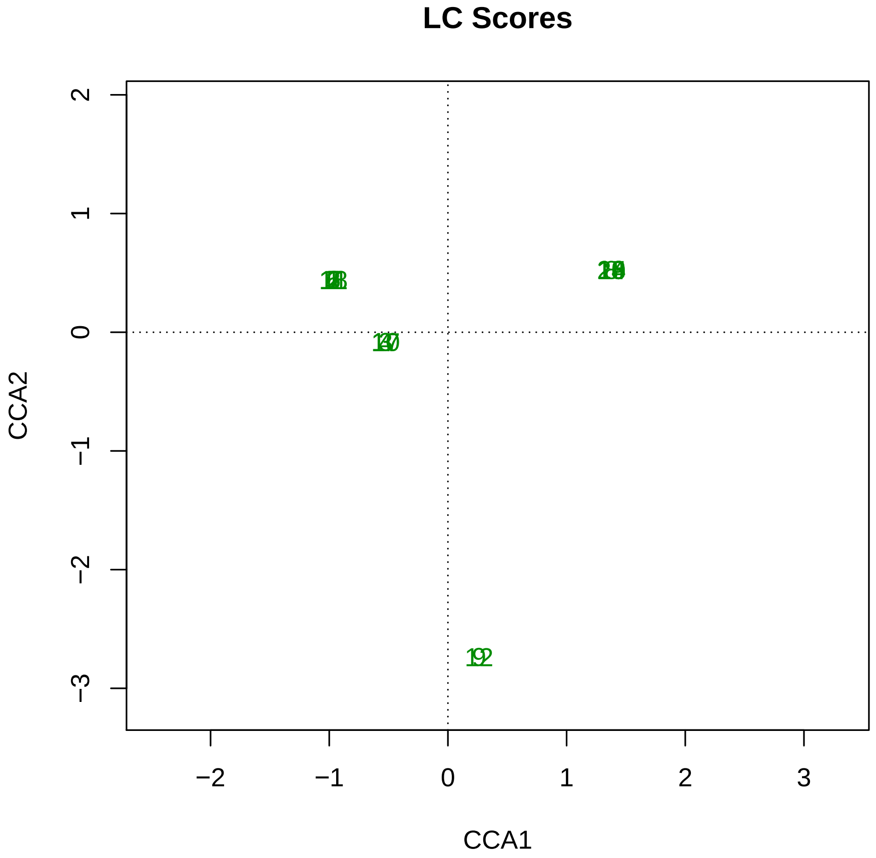
Bruce McCune:

- LC scores are excellent, if you have no error in constraining variables. Even with small error, LC scores become miserable, but WA scores are good even in noisy data.



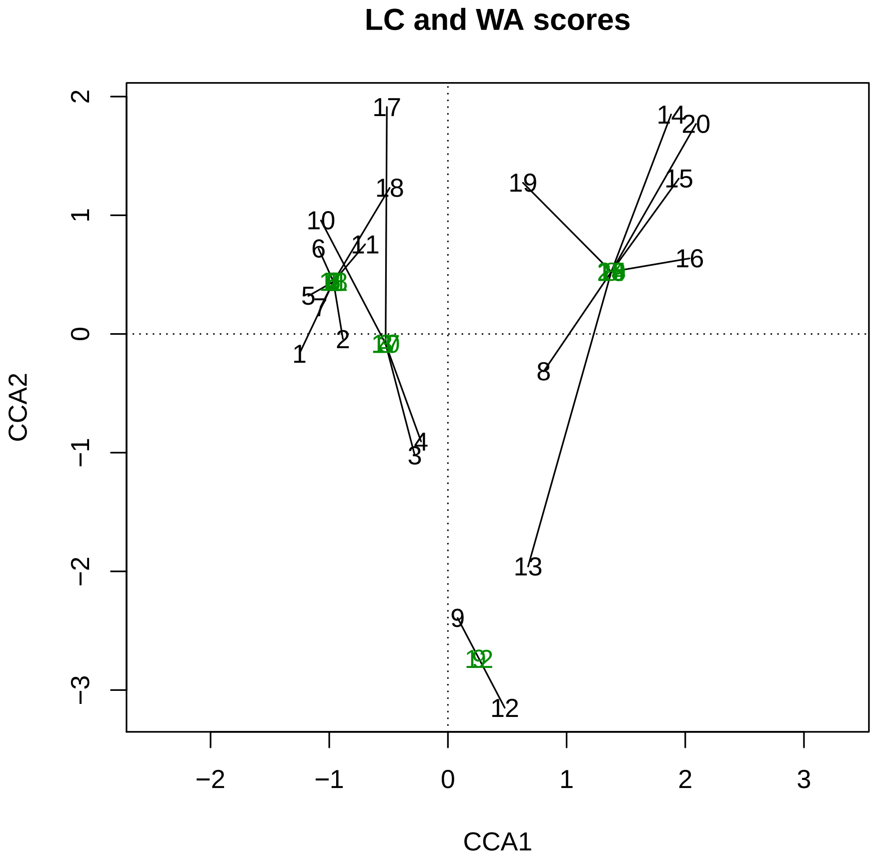
LC Scores are Constraints

Dune Meadows Constrained by Moisture Level



LC Scores are Constraints

Dune Meadows Constrained by Moisture Level



Outline

1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis
2. Analysis of Dissimilarities
   - Methods

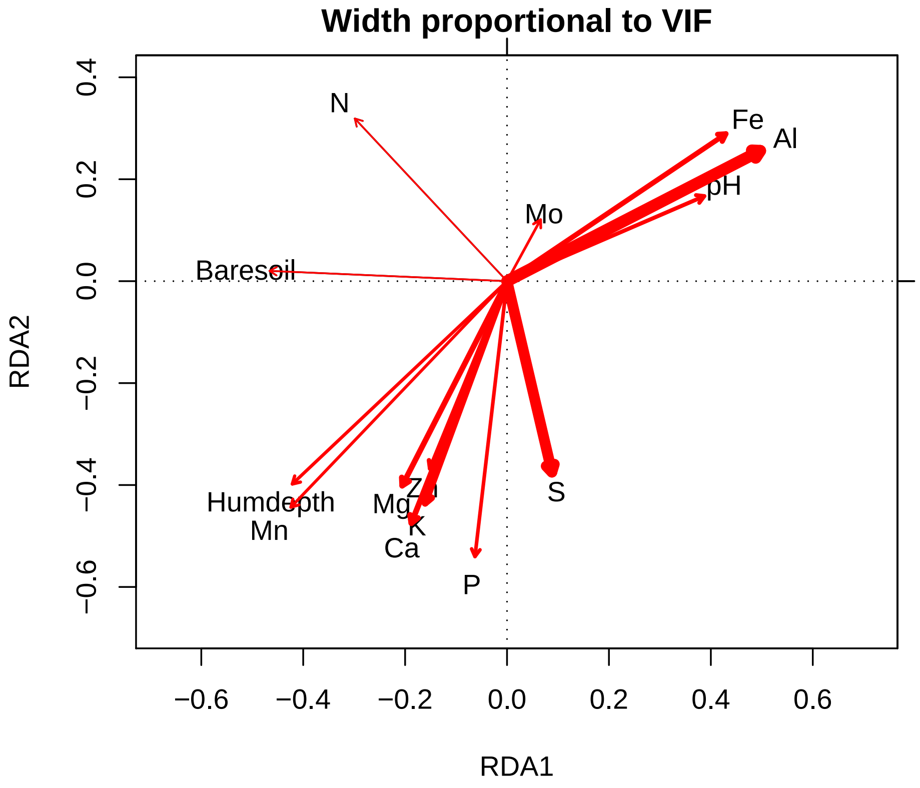
Model Choice

- Often people chunk in all environmental variables they have – a patently bad idea
- Increasing the number of constraints means slacker constraint: analysis approaches unconstrained ordination and fitting environmental variables
- Does not allow hypothesis testing
- Many of the variables may be insignificant
- Multicollinearity between variables evident as *Variance Inflation Factor* (VIF)

```
> vif.cca(cca(varespec, varechem))
```

```
       N        P        K       Ca       Mg        S       Al
    1.98     6.03    12.01     9.93     9.81    18.38    21.19
      Fe       Mn       Zn       Mo Baresoil Humdepth       pH
    9.13     5.38     7.74     4.32     2.25     6.01     7.39
```

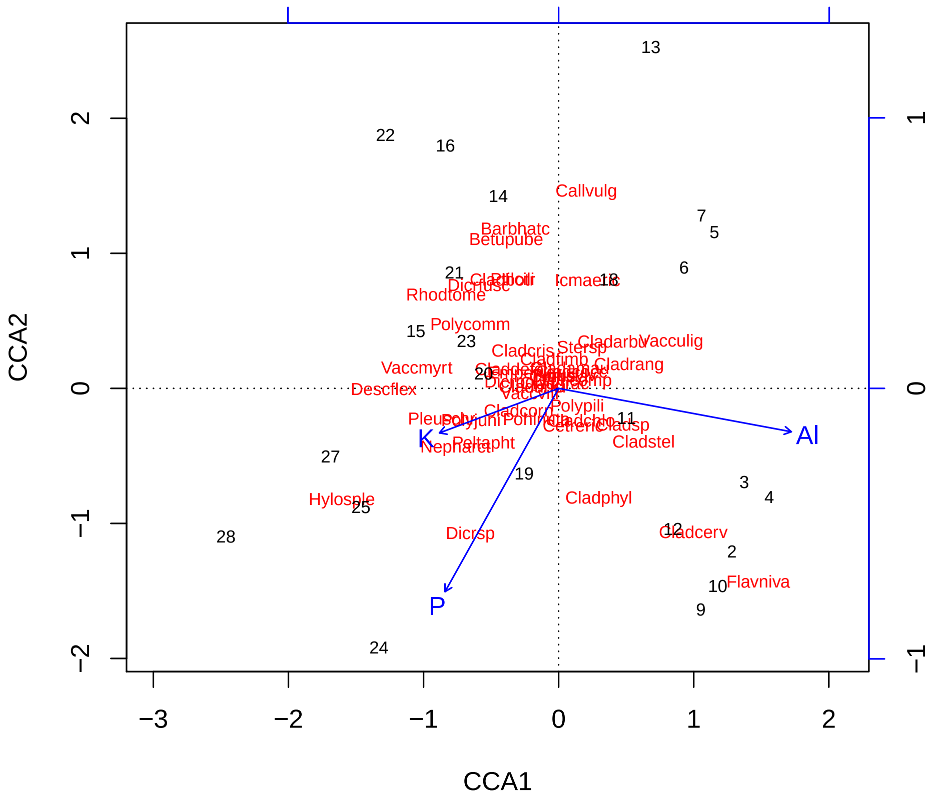
Variance Inflation Factor



Model Specification: Formula Interface I

```
> (vare.cca <- cca(varespec ~ Al + P + K, varechem))

Call: cca(formula = varespec ~ Al + P + K, data =
varechem)

              Inertia Proportion Rank
Total           2.083      1.000
Constrained     0.644      0.309    3
Unconstrained   1.439      0.691   20
Inertia is mean squared contingency coefficient

Eigenvalues for constrained axes:
 CCA1  CCA2  CCA3
0.362 0.170 0.113

Eigenvalues for unconstrained axes:
  CA1   CA2   CA3   CA4   CA5   CA6   CA7   CA8
0.350 0.220 0.185 0.155 0.135 0.100 0.077 0.054
(Showed only 8 of all 20 unconstrained eigenvalues)

> vif.cca(vare.cca)

  Al    P    K
1.01 2.37 2.38
```

Plot



Coding Factors

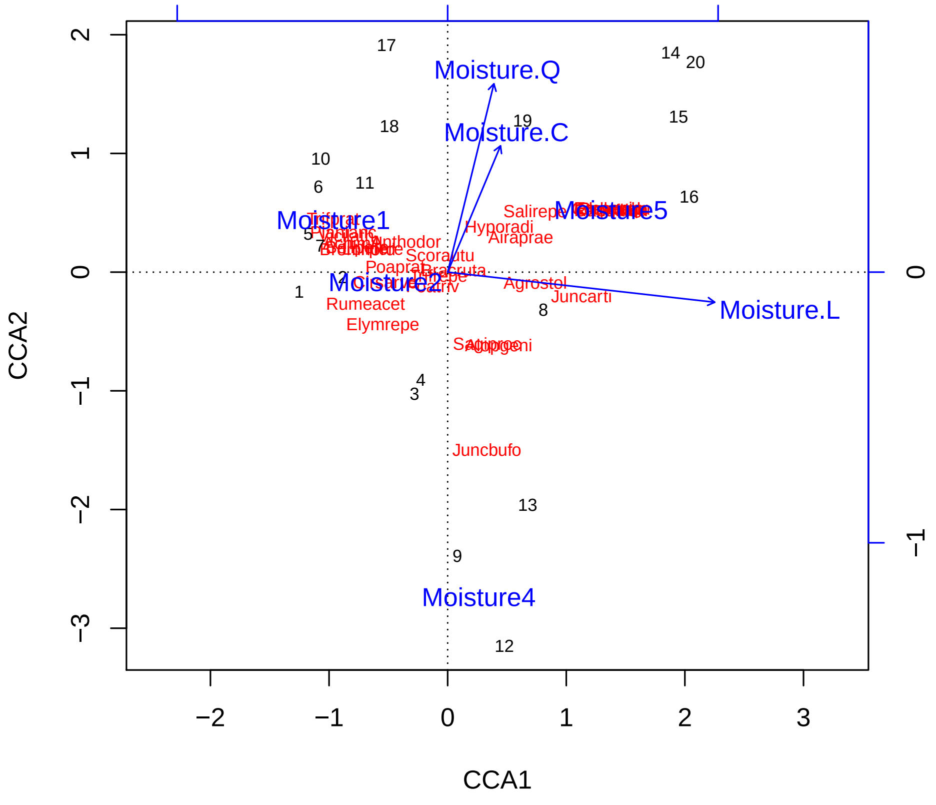
Dummy Variables:

```
   ManagementHF ManagementNM ManagementSF
SF            0            0            1
BF            0            0            0
HF            1            0            0
NM            0            1            0
```

Ordered Factors:

```
  Moisture.L Moisture.Q Moisture.C
1     -0.671        0.5     -0.224
2     -0.224       -0.5      0.671
4      0.224       -0.5     -0.671
5      0.671        0.5      0.224
```

Plotting Ordered Factors



Goodness of Model and its Costs

- Eigenvalue is the measure of goodness of fit
- Eigenvalue is maximized: even random constraints will have $\lambda > 0$, and eigenvalues will grow when you add constraints
- AIC: balance eigenvalue by a penalty for each used constraint
- AIC does not exist for constrained ordination: AIC is based on Likelihood of the fitted model, and ordination models do not have Likelihood
- Toy-AIC may sometimes work, and can be used in automated model building
- Permutation tests can be used to check the approximate validity of automated model building

Shortcut to a Maximal Model I

```
> mod1 <- cca(varespec ~ ., varechem)
> mod1

Call: cca(formula = varespec ~ N + P + K + Ca + Mg + S
+ Al + Fe + Mn + Zn + Mo + Baresoil + Humdepth + pH,
data = varechem)

              Inertia Proportion Rank
Total           2.083      1.000
Constrained     1.441      0.692   14
Unconstrained   0.642      0.308    9
Inertia is mean squared contingency coefficient

Eigenvalues for constrained axes:
 CCA1  CCA2  CCA3  CCA4  CCA5  CCA6  CCA7  CCA8  CCA9 CCA10
0.439 0.292 0.163 0.142 0.118 0.089 0.070 0.058 0.031 0.013
CCA11 CCA12 CCA13 CCA14
0.008 0.007 0.006 0.005

Eigenvalues for unconstrained axes:
   CA1    CA2    CA3    CA4    CA5    CA6    CA7    CA8    CA9
0.1978 0.1419 0.1012 0.0708 0.0533 0.0333 0.0189 0.0151 0.0095
```

Stepping to a Good Model I

```
> mod0 <- cca(varespec ~ 1, varechem)
> mod <- step(mod0, scope = formula(mod1), test="perm", perm.max=100)
```

```
Start:  AIC=130.31
varespec ~ 1

           Df    AIC      F Pr(>F)
+ Al        1 128.61 3.6749  0.005 **
+ Mn        1 128.95 3.3115  0.005 **
+ Humdepth  1 129.24 3.0072  0.005 **
+ Baresoil  1 129.77 2.4574  0.035 *
+ Fe        1 129.79 2.4360  0.020 *
+ P         1 130.03 2.1926  0.025 *
+ Zn        1 130.30 1.9278  0.060 .
<none>        130.31
+ Mg        1 130.35 1.8749  0.045 *
+ K         1 130.37 1.8609  0.060 .
+ Ca        1 130.43 1.7959  0.070 .
+ pH        1 130.57 1.6560  0.115
+ S         1 130.72 1.5114  0.135
+ N         1 130.77 1.4644  0.135
+ Mo        1 131.19 1.0561  0.400
```

Stepping to a Good Model II

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=128.61
varespec ~ Al

           Df    AIC      F Pr(>F)
+ P         1 127.91 2.5001  0.010 **
+ K         1 128.09 2.3240  0.015 *
+ S         1 128.26 2.1596  0.025 *
+ Zn        1 128.44 1.9851  0.030 *
+ Mn        1 128.53 1.8945  0.025 *
<none>        128.61
+ Mg        1 128.70 1.7379  0.055 .
+ N         1 128.85 1.5900  0.095 .
+ Baresoil  1 128.88 1.5670  0.135
+ Ca        1 129.04 1.4180  0.160
+ Humdepth  1 129.08 1.3814  0.210
+ Mo        1 129.50 0.9884  0.465
+ pH        1 129.63 0.8753  0.575
+ Fe        1 130.02 0.5222  0.860
- Al        1 130.31 3.6749  0.005 **
```

Stepping to a Good Model III

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=127.91
varespec ~ Al + P

           Df    AIC      F Pr(>F)
+ K         1 127.44 2.1688  0.040 *
<none>        127.91
+ Baresoil  1 127.99 1.6606  0.090 .
+ N         1 128.11 1.5543  0.140
+ S         1 128.36 1.3351  0.225
+ Mn        1 128.44 1.2641  0.235
+ Zn        1 128.51 1.2002  0.330
+ Humdepth  1 128.56 1.1536  0.360
- P         1 128.61 2.5001  0.015 *
+ Mo        1 128.75 0.9837  0.450
+ Mg        1 128.79 0.9555  0.465
+ pH        1 128.82 0.9247  0.460
+ Fe        1 129.28 0.5253  0.875
+ Ca        1 129.36 0.4648  0.910
- Al        1 130.03 3.9401  0.005 **
```

Stepping to a Good Model IV

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=127.44
varespec ~ Al + P + K

           Df    AIC      F Pr(>F)
<none>        127.44
+ N         1 127.59 1.5148  0.135
+ Baresoil  1 127.67 1.4544  0.145
+ Zn        1 127.84 1.3067  0.185
+ S         1 127.89 1.2604  0.265
- K         1 127.91 2.1688  0.005 **
+ Mo        1 127.92 1.2350  0.225
- P         1 128.09 2.3362  0.010 **
+ Mg        1 128.17 1.0300  0.385
+ Mn        1 128.34 0.8879  0.490
+ Humdepth  1 128.44 0.8056  0.660
+ Fe        1 128.79 0.5215  0.830
+ pH        1 128.81 0.5067  0.880
+ Ca        1 128.89 0.4358  0.895
- Al        1 130.14 4.3340  0.005 **
```

Stepping to a Good Model V

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> mod

Call: cca(formula = varespec ~ Al + P + K, data =
varechem)

              Inertia Proportion Rank
Total          2.0832     1.0000
Constrained    0.6441     0.3092    3
Unconstrained  1.4391     0.6908   20
Inertia is mean squared contingency coefficient

Eigenvalues for constrained axes:
  CCA1   CCA2   CCA3
0.3616 0.1700 0.1126

Eigenvalues for unconstrained axes:
   CA1    CA2    CA3    CA4    CA5    CA6    CA7    CA8
0.3500 0.2201 0.1851 0.1551 0.1351 0.1003 0.0773 0.0537
(Showed only 8 of all 20 unconstrained eigenvalues)
```

Other Methods of Model Choice

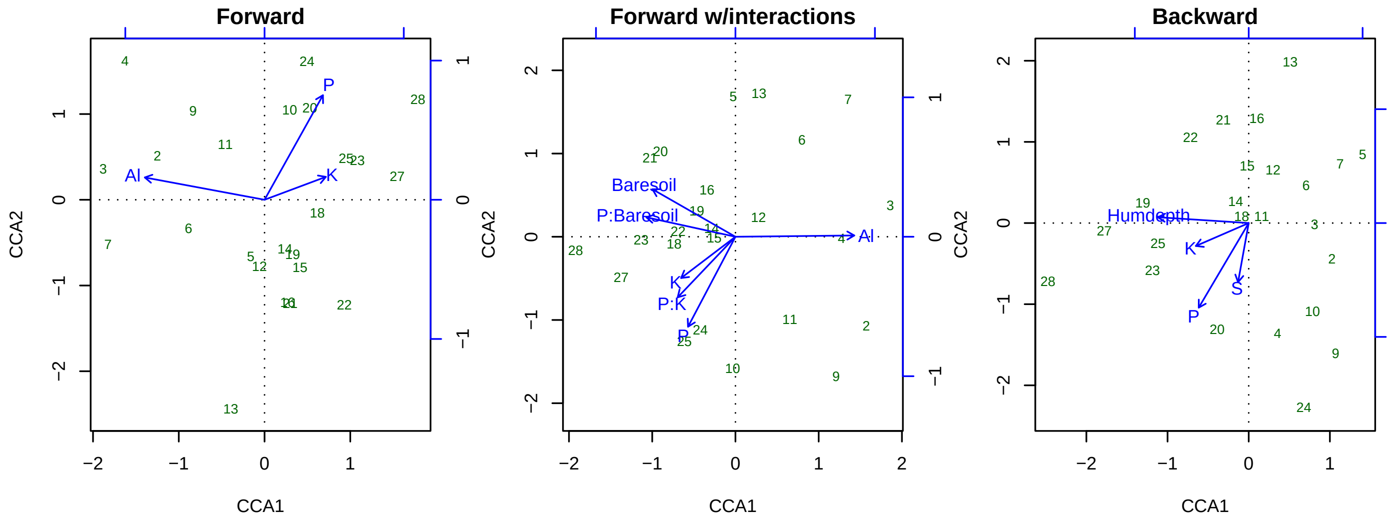
- Selection of terms by permutation tests (ordistep)
  - Ties broken by pseudo-AIC
  - Inclusion limit defaults $P = 0.05$ and exclusion limit $P = 0.1$
- Select terms to maximize adjusted R^2_{adj} (ordiR2step)
  - adjusted R^2 is penalized by the number of constraints p and can decrease when terms are added

$$R^2_{\text{adj}} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$$

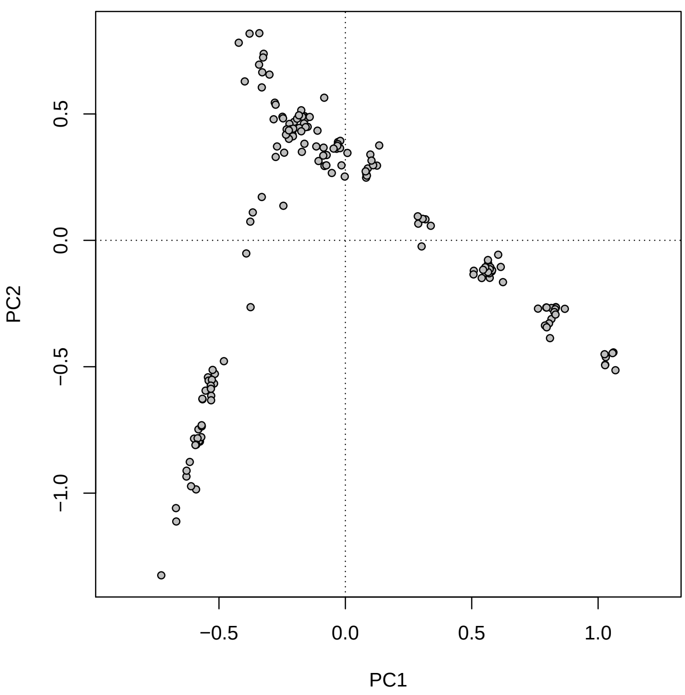
  - The expected value $R^2_{\text{adj}} = 0$ in random data, but the expected value for unadjuted $R^2 > 0$
  - Adjusted R^2 is only available for Euclidean methods (RDA, db-RDA), but not for CCA
  - Other stopping criteria: R^2_{adj} exceeds that of the full model, or terms are deemed insignificant by permutation tests

Stepping is Dangerous

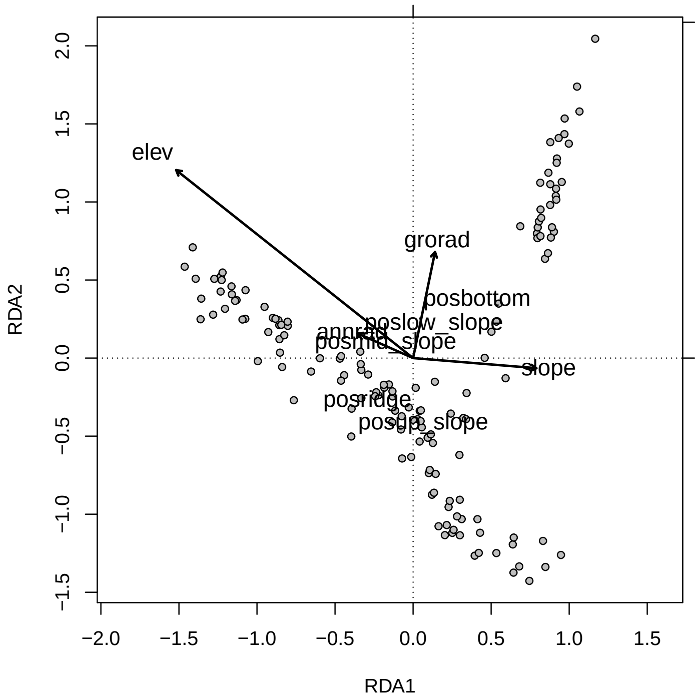
Automatic model selection may give different results depending on stepping direction, scope or small changes in the data set



Ordination Wants to be Free!

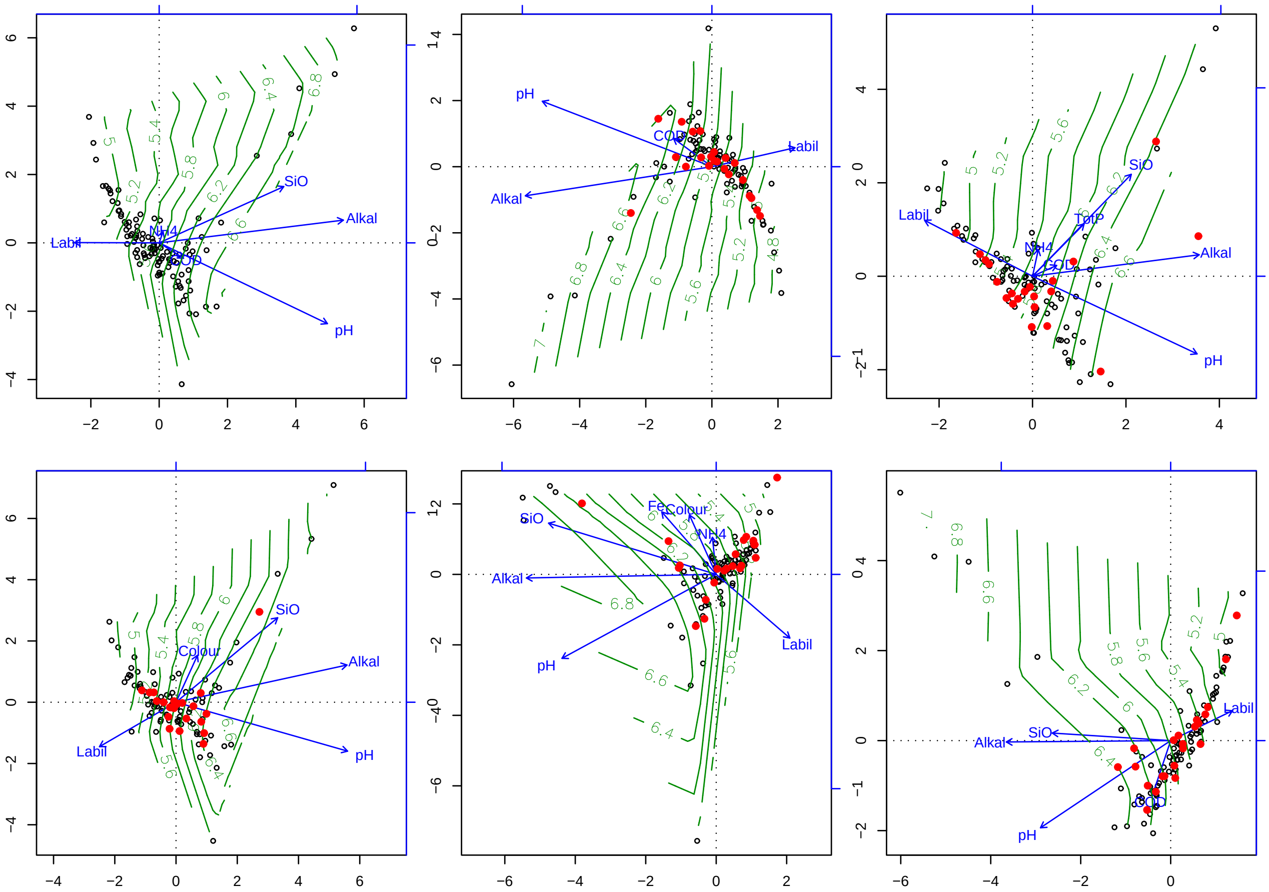


Unconstrained ordination (Bryce Canyon)



Automatic model building: the return of the curve

5-fold Cross-validation and stepping



Outline

1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis

2. Analysis of Dissimilarities
   - Methods

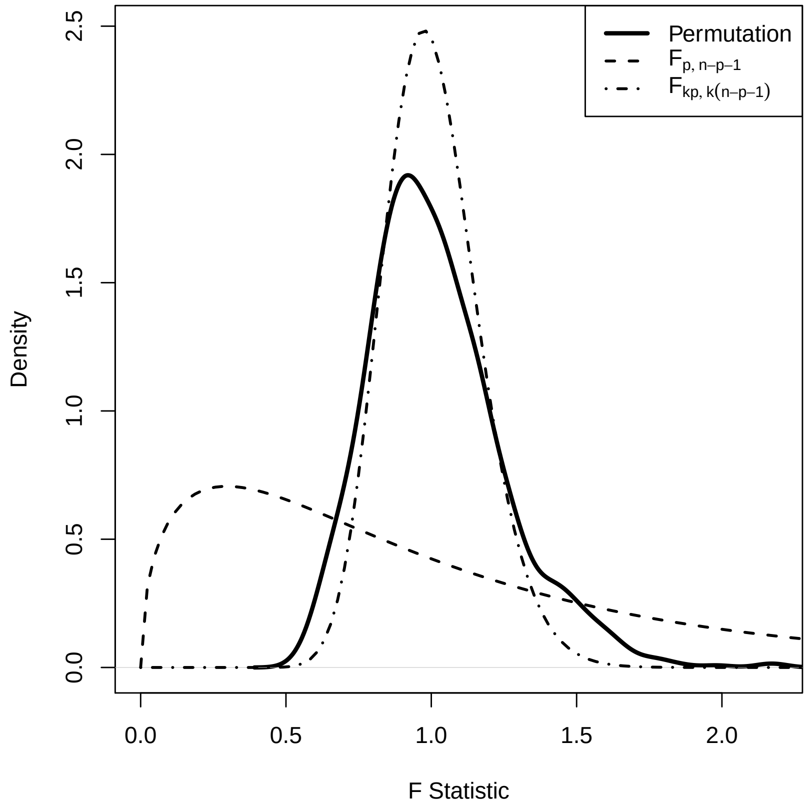
Permutation Test

- The significance of constraints cannot be directly evaluated, but we can use permutation tests
- Shuffle community data into random order and refit the model: gives goodness of fit of a random model
- If observed goodness of fit is better than (most) random models, then the constraints are significant
- The observed goodness could be just one of the random values, and it is put together with permutations: for nice divisor of 1000 we generate 999 permutations and divide with $999 + 1$
- The criterion of the goodness of fit is pseudo-F:

  $$F = \frac{\Lambda_c/p}{\Lambda_r/(n-p-1)},$$

  where Λ_c and Λ_r are constrained and residual inertia (and total inertia $\Lambda = \Lambda_c + \Lambda_r$), p is the rank of constraints, and n is the number of observations
- Definition similar to F-statistic in ANOVA, but does not follow its distribution (except for single variable in RDA)

Distribution of the Statistic



No. of observations n, no. of species k, no. of constraints p

Overall Test

```
> anova(mod)

Permutation test for cca under reduced model
Permutation: free
Number of permutations: 999

Model: cca(formula = varespec ~ Al + P + K, data = varechem)
         Df ChiSquare    F Pr(>F)
Model     3     0.644 2.98  0.001 ***
Residual 20     1.439
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

ANOVA by Terms

```
> anova(mod, by=”terms”)

Permutation test for cca under reduced model
Terms added sequentially (first to last)
Permutation: free
Number of permutations: 999

Model: cca(formula = varespec ~ Al + P + K, data = varechem)
         Df ChiSquare    F Pr(>F)
Al        1     0.298 4.14  0.001 ***
P         1     0.190 2.64  0.008 **
K         1     0.156 2.17  0.017 *
Residual 20     1.439
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

ANOVA by Margins

Type III Sums of Squares

```
> anova(mod, by=”mar”)

Permutation test for cca under reduced model
Marginal effects of terms
Permutation: free
Number of permutations: 999

Model: cca(formula = varespec ~ Al + P + K, data = varechem)
         Df ChiSquare    F Pr(>F)
Al        1     0.312 4.33  0.001 ***
P         1     0.168 2.34  0.019 *
K         1     0.156 2.17  0.027 *
Residual 20     1.439
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

ANOVA by Axis

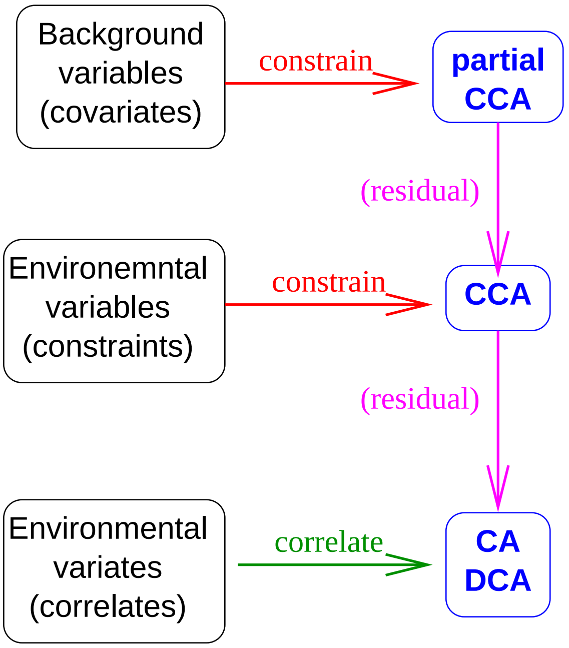
```
> anova(vare.cca, by=”axis”, perm=1000)

Permutation test for cca under reduced model
Marginal tests for axes
Permutation: free
Number of permutations: 999

Model: cca(formula = dune ~ Moisture, data = dune.env)
         Df ChiSquare    F Pr(>F)
CCA1      1     0.419 4.51  0.001 ***
CCA2      1     0.133 1.43  0.119
CCA3      1     0.077 0.82  0.602
Residual 16     1.487
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Outline

1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis

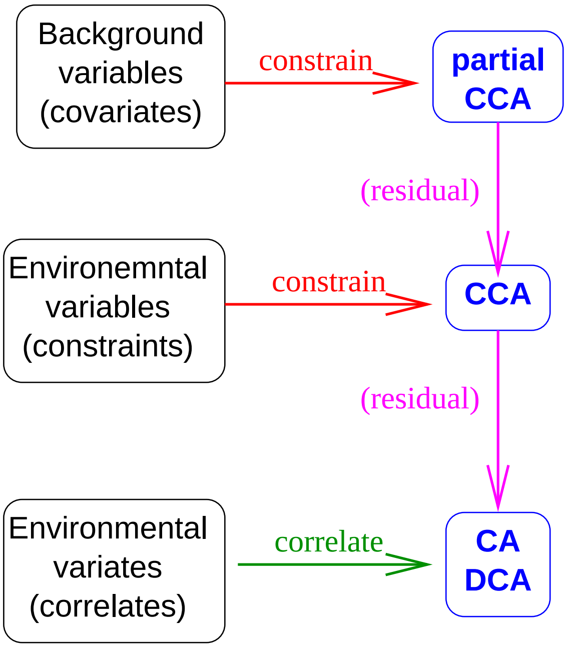
2. Analysis of Dissimilarities
   - Methods

Levels of Intervention


Background variables (covariates)
constrain
partial CCA
(residual)
Environemntal variables (constraints)
constrain
CCA
(residual)
Environmental variates (correlates)
correlate
CA
DCA

- **Partial CCA** removes the effect of background variables before proper (C)CA: ‘random’ or ‘nuisance’ variables.
- Residual ordinations: Partitioning of variation.

Levels of Intervention


Background variables (covariates)
constrain
partial CCA
(residual)
Environemntal variables (constraints)
constrain
CCA
(residual)
Environmental variates (correlates)
correlate
CA DCA

- **Partial CCA** removes the effect of background variables before proper (C)CA: ‘random’ or ‘nuisance’ variables.
- Residual ordinations: Partitioning of variation.
- Constraints are linear: Non-orthogonal environmental variables may give ‘negative components of variation’
- Information of lower levels mixed with upper.

Why Partial Ordination?

- Remove the effect of background (or “random”) variables before analysing the effect of interesting variables
- Allows analysis of experimental design (constraints) with confounding variables (conditions)
- Allows split-plot and other hierarchical designs
- Decomposition of variation due to different sources, like spatial and environmental components

Treatment with Confounding Natural Variation I

```
> (ord <- rda(dune ~ Management + Condition(A1 + Moisture), dune.env))

Call: rda(formula = dune ~ Management + Condition(A1 +
Moisture), data = dune.env)

              Inertia Proportion Rank
Total          84.124      1.000
Conditional    29.765      0.354    4
Constrained    19.115      0.227    3
Unconstrained  35.244      0.419   12
Inertia is variance

Eigenvalues for constrained axes:
 RDA1  RDA2  RDA3
11.26  4.88  2.97

Eigenvalues for unconstrained axes:
 PC1  PC2  PC3  PC4  PC5  PC6  PC7  PC8  PC9 PC10 PC11 PC12
8.21 7.14 4.61 4.03 3.02 2.66 1.87 1.50 0.91 0.64 0.39 0.27

> anova(ord)
```

Treatment with Confounding Natural Variation II

```
Permutation test for rda under reduced model
Permutation: free
Number of permutations: 999

Model: rda(formula = dune ~ Management + Condition(A1 + Moisture), data = dune.env)
         Df Variance    F Pr(>F)
Model     3     19.1 2.17  0.004 **
Residual 12     35.2
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

What Actually is Permuted in Tests?

- **Direct Model**: Always permutes community data
- **Reduced Model**: Permutes community data in non-partial models, and residuals after conditions in partial model
- When residuals are permuted in reduced model, the permuted residuals are added to the unpermuted fitted values
- Theory assume that residuals are *exchangeable*, and hypothesis of randomness concern residuals
- Assumes *independent and identically distributed* residuals: these can be added to fitted values

Components of Variation

- There can be several groups of source of variation, and we may be interested in quantifying these components
- Typical example: decomposition of variation into pure spatial, pure environmental and spatially structured environmental variation
- We expect that usual $R^2 > 0$, because the goodness of fit is maximized, but adjusted R^2 takes into account the number of constraints and has expectation 0 with random constraints
- Spatial structure can be described by Principal Components of Neighbourhood Matrix (PCNM)

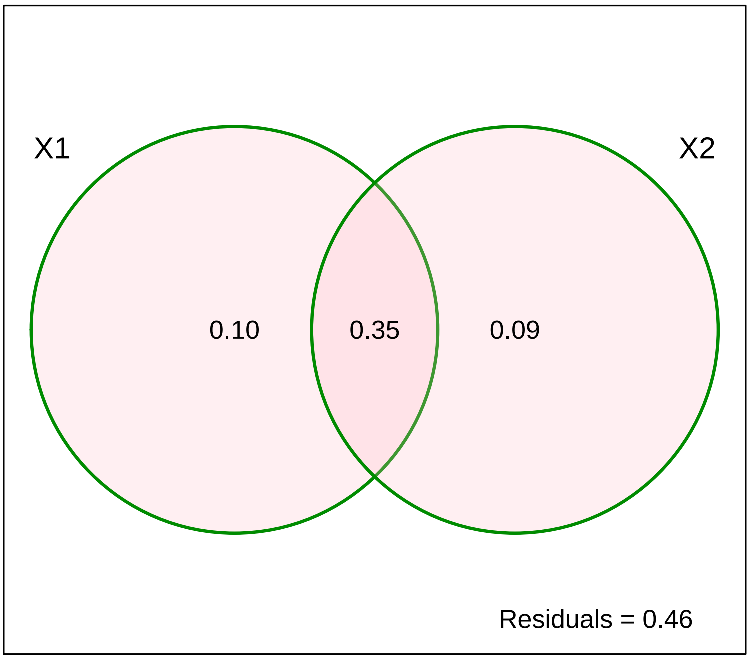
Example: Spatial and Environmental Variation I

```
> (mod <- varpart(mite, mite.pcnm,  ~. , data=mite.env, transfo="hellinger"))

Partition of variance in RDA

Call: varpart(Y = mite, X = mite.pcnm, ~., data =
mite.env, transfo = "hellinger")
Species transformation:  hellinger
Explanatory tables:
X1:  mite.pcnm
X2:  ~.

No. of explanatory tables: 2
Total variation (SS): 27.205
            Variance: 0.39428
No. of observations: 70

Partition table:
                     Df R.squared Adj.R.squared Testable
[a+b] = X1           22   0.62300       0.44653     TRUE
[b+c] = X2           11   0.52650       0.43670     TRUE
[a+b+c] = X1+X2      33   0.75893       0.53794     TRUE
Individual fractions
```

Example: Spatial and Environmental Variation II

```
[a] = X1|X2          22                 0.10124     TRUE
[b]                   0                 0.34530    FALSE
[c] = X2|X1          11                 0.09141     TRUE
[d] = Residuals                         0.46206    FALSE
---
Use function ‘rda’ to test significance of fractions of interest
```

Components of Variance



Outline

1. Constrained Ordination
   - Methods
   - Model Choice
   - Permutation Test
   - Partial Analysis
2. Analysis of Dissimilarities
   - Methods

Direct Analysis of Dissimilarities

- Analyse dissimilarities instead of mapping them into reduced number of dimensions of ordination
- Distance-based Redundancy Analysis (capscale in vegan) can perform the reduction
- Want to have non-Euclidean metric?
- Want to study the effect of geographic (spatial) distance?
- Do you have huge number of variables, but a modest number of observations (like in genetic data)

Distance-based RDA I

```
> pcnmmat <- as.matrix(mite.pcnm)
> (ord <- capscale(vegdist(mite)  ~ . + pcnmmat, mite.env))

Call: capscale(formula = vegdist(mite) ~ SubsDens +
WatrCont + Substrate + Shrub + Topo + pcnmmat, data =
mite.env)

              Inertia Proportion Eigenvals Rank
Total          14.696      1.000    16.742     
Constrained    10.968      0.746    11.902   33
Unconstrained   3.728      0.254     4.840   36
Imaginary                           -2.046   32
Inertia is squared Bray distance

Eigenvalues for constrained axes:
 CAP1  CAP2  CAP3  CAP4  CAP5  CAP6  CAP7  CAP8  CAP9 CAP10
 5.24  1.46  1.12  0.75  0.56  0.44  0.36  0.26  0.25  0.23
CAP11 CAP12 CAP13 CAP14 CAP15 CAP16 CAP17 CAP18 CAP19 CAP20
 0.19  0.18  0.14  0.12  0.10  0.09  0.07  0.07  0.05  0.05
CAP21 CAP22 CAP23 CAP24 CAP25 CAP26 CAP27 CAP28 CAP29 CAP30
 0.04  0.03  0.03  0.02  0.02  0.01  0.01  0.01  0.01  0.00
CAP31 CAP32 CAP33
```

Distance-based RDA II

```
 0.00  0.00  0.00

Eigenvalues for unconstrained axes:
 MDS1  MDS2  MDS3  MDS4  MDS5  MDS6  MDS7  MDS8
1.063 0.597 0.372 0.354 0.327 0.290 0.275 0.202
(Showed only 8 of all 36 unconstrained eigenvalues)
```

```
> anova(ord, by="margin", perm.max=1000)
```

```
Permutation test for capscale under reduced model
Marginal effects of terms
Permutation: free
Number of permutations: 999

Model: capscale(formula = vegdist(mite) ~ SubsDens + WatrCont + Substrate + Shrub + Topo +
          Df SumOfSqs    F Pr(>F)
SubsDens   1     0.10 1.00  0.401
WatrCont   1     0.27 2.59  0.032 *
Substrate  6     0.94 1.52  0.052 .
Shrub      2     0.08 0.40  0.968
Topo       1     0.16 1.58  0.142
pcnmmat   22     3.54 1.56  0.003 **
Residual  36     3.73
```

Distance-based RDA III

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Mantel and Partial Mantel Tests

- Mantel correlation (a.k.a. matrix correlation) is the correlation between two sets of dissimilarities or distances
- $n(n-1)/2$ dissimilarities for n independent observations: ordinary statistical tests do not apply
- Significance can be assessed by permutations
- Partial Mantel test: use three sets of dissimilarities and partial correlations conditioning relationship between two sets by the third one
- Analogous to conditioned db-RDA: partial out variation by background distances
- Residuals of distances are not equivalent to residuals of raw data: decomposition of variation dubious

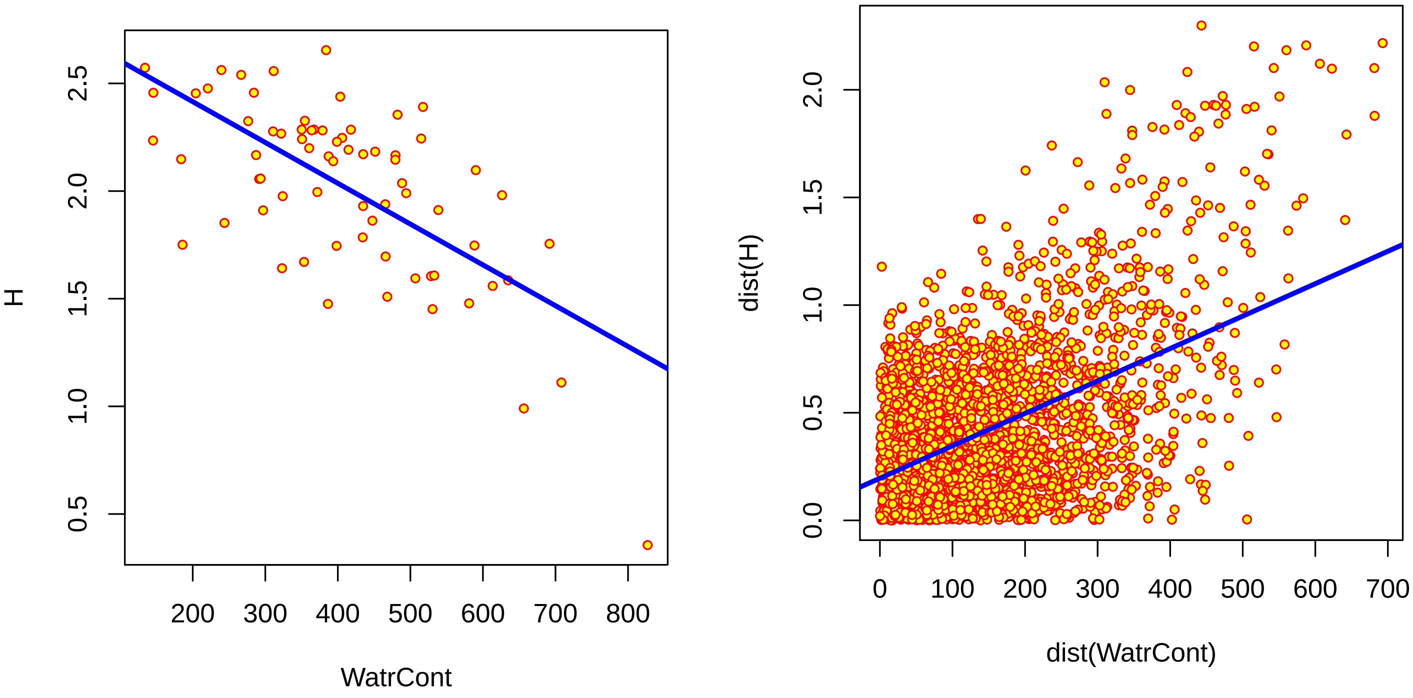
Example: Community Structure and Environment

```
> library(cluster)
> envdis <- daisy(mite.env)
> mantel(vegdist(mite), envdis)

Mantel statistic based on Pearson's product-moment correlation

Call:
mantel(xdis = vegdist(mite), ydis = envdis)

Mantel statistic r: 0.422
      Significance: 0.001

Upper quantiles of permutations (null model):
   90%    95%  97.5%    99%
0.0417 0.0528 0.0624 0.0762
Permutation: free
Number of permutations: 999
```

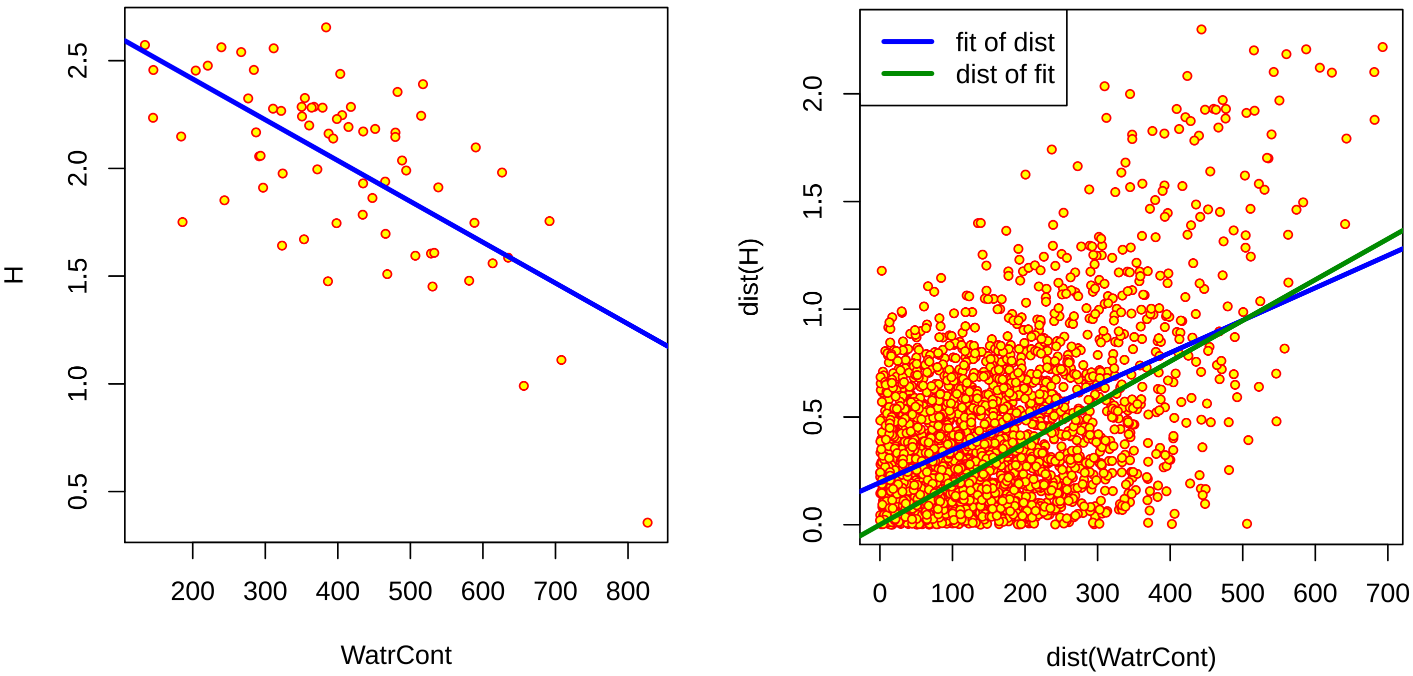
Controlling for Spatial Distance

```
> mantel.partial(vegdist(mite), envdis, dist(mite.xy))

Partial Mantel statistic based on Pearson's product-moment correlation

Call:
mantel.partial(xdis = vegdist(mite), ydis = envdis, zdis = dist(mite.xy))

Mantel statistic r: 0.292
      Significance: 0.001

Upper quantiles of permutations (null model):
   90%    95%  97.5%    99%
0.0416 0.0562 0.0635 0.0753
Permutation: free
Number of permutations: 999
```

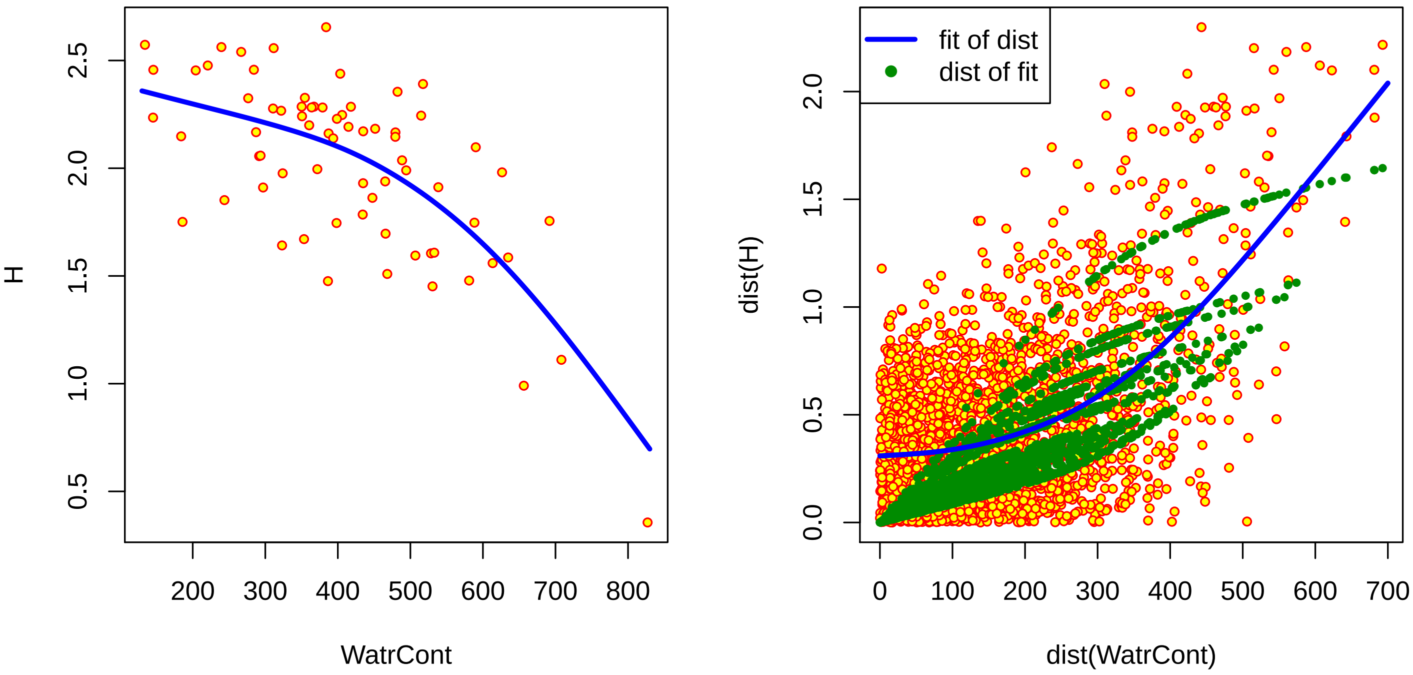
Direct Way and Mantel Way



Direct Way and Mantel Way



Direct Way and Mantel Way



Linear Analysis of Dissimilarities

- Function adonis in vegan
- Permutational MANOVA or non-parametric MANOVA
- Uses “outer products” in MANOVA instead of usual “inner products”: dissimilarities among points instead of distances of variables to their centroids
- Does not use raw distances, but transforms them to principal coordinates for a “direct analysis”: usually more powerful than Mantel style
- Practical if the number of variables is huge: related to AMOVA of gene expression data
- With Euclidean distances equal to MANOVA, but uses permutation tests
- Can be used with any adequate dissimilarity measure
- Test sequential: order of variables does matter

Example: Environment after Spatial Variation

```
> adonis(vegdist(mite)  ~ pcnmmat + ., mite.env, perm=500)

Call:
adonis(formula = vegdist(mite) ~ pcnmmat + ., data = mite.env,      permutations = 500)

Permutation: free
Number of permutations: 500

Terms added sequentially (first to last)

          Df SumsOfSqs MeanSqs F.Model    R2 Pr(>F)
pcnmmat   22      8.84   0.402    3.88 0.601  0.002 **
SubsDens   1      0.41   0.410    3.96 0.028  0.008 **
WatrCont   1      0.32   0.324    3.13 0.022  0.014 *
Substrate  6      1.07   0.179    1.73 0.073  0.018 *
Shrub      2      0.16   0.080    0.77 0.011  0.663
Topo       1      0.16   0.164    1.58 0.011  0.138
Residuals 36      3.73   0.104         0.254
Total     69     14.70                 1.000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Other Dissimilarity-based Methods

- MRPP (Multi-Response Permutation Procedure) and ANOSIM (Analysis of Dissimilarities) compare differences among groups
  - Both are sensitive to differences in the dispersions within groups: **not recommended**
- Multivariate analysis of homogeneity (`betadisper` in vegan)
  - With Euclidean distances equal to Levene's test on the homegeneity of variances
  - Also works exactly on non-Euclidean dissimilarities
  - Can be used to study beta diversity within groups
  - Either parametric ANOVA or permutation tests available
  - Pairwise *post hoc* comparison available (Tukey)
  - PERMDISP2 by another name