1. Supporting Information

Testing environmental effects on taxonomic composition with canonical correspondence analysis: alternative permutation tests are not equal

Cajo J. F. ter Braak[1] and Dennis E. te Beest[1]

[1] Biometris, Wageningen University & Research, Wageningen, the Netherlands

Appendices

The software used in this paper (R-code) and the appendices are available at https://doi.org10.6084/m9.figshare.15016008. See also https://doi.org/10.6084/m9.figshare.13259534

1. Gaussian response and log-linear models for CCA

In this appendix we show in two ways that the Gaussian response model is closely linked to a particular log-linear model known as the Goodman (1986) RC-model, of which we consider the constrained version. The first way is exact and hinges on the addition of free site parameters to the Gaussian model that make it suited for multinomial and compositional data. The second way is approximate and holds true for small effects of the environmental variables on the species abundance. We then show how CCA is related to this particular log-linear model by showing that the transition formulas of CCA are an approximation to the ML-equation of this log-linear model under the assumption that effects are small and the abundances are Poisson distributed. We conclude with some remarks.

1.1 From Gaussian response to log-linear models

We start from the Gaussian model for a single environmental variable with formula

$$\mu_{ij} = E(y_{ij}) = r_i^* c_j^* e^{-(x_i-u_j)^2/2t_j^2}, \qquad \text{(A1)}$$

where μ_{ij} and y_{ij} denote the expected and observed abundance of species j in site i, respectively ($i = 1,\ldots, n; j = 1, \ldots, m$), E(.) denotes expectation and x_i is the value of the environmental variable in site i. The unknown parameters c_j^*, u_j and t_j are the maximum expected abundance, optimum and tolerance of species j, respectively, and r_i^* is an

unknown site parameter that may account of unobserved variation in sampling effort among sites. The traditional model used in ecology is obtained by setting $r_i^* = 1$. With r_i^* included in the estimation, the model is equivalent to the generalized logistic model ("model B") of Ihm and van Groenewoud (1984); see also ter Braak (1988).

Expanding the square in equation (A1) gives

$$\mu_{ij} = r_i^* c_j^* e^{u_j x_i/t_j^2 - x_i^2/2t_j^2 - u_j^2/2t_j^2}. \quad \text{(A2)}$$

The term $e^{-u_j^2/2t_j^2}$ can be absorbed in the parameter c_j^*. Under the assumption of equal tolerance across species ($t_j = t$), the term $e^{-x_i^2/2t_j^2}$ can be absorbed in the site parameter r_i^*. With $R_i = r_i^* e^{-x_i^2/2t^2}$, $C_j = c_j^* e^{-u_j^2/2t_j^2}$, the resulting model is

$$\mu_{ij} = R_i C_j e^{u_j x_i/t^2} = R_i C_j e^{b_j x_i} \text{ where } b_j = u_j/t^2. \quad \text{(A3)}$$

This model is known as the (constrained) RC (for row-column) model of Goodman (1986) and can be expressed as a log-linear model

$$log\, \mu_{ij} = \tilde{r}_i + \tilde{c}_j + b_j x_i \quad \text{(A4)}$$

by setting $\tilde{r}_i = \log(R_i)$ and $\tilde{c}_j = \log(C_j)$. This first derivation required the assumption of equal tolerance ($t_j = t$), which may not be valid and may not be needed in the context of hypothesis testing.

For the second derivation, we consider the case that the species respond approximately monotonic to the environmental variable in the range of values of the environmental variable **x**. This can be achieved in equation (A1) by increasing the tolerance of all species ($t_j \to \infty$) and also increasing the optima in absolute value ($|u_j| \to \infty$), in such a way that $b_j = u_j/t_j^2$ is finite for all values of j. With $R_i = r_i^*$ and $C_j = c_j^* e^{-u_j^2/2t_j^2}$, the expanded Gaussian model (A2) then becomes

$$\mu_{ij} \approx R_i C_j e^{b_j x_i}, \quad \text{(A5)}$$

so that it is approximately equal to the RC-model (A3) or its log-linear equivalent (A4).

1.2 From the log-linear model to CCA

For hypothesis testing, we are specifically interested in the case with small values of b_j, as such testing wishes to distinguish between no effect and non-zero (small) effects for at least one species.

1.1.1 The reconstitution formula of CCA when effects are small

If b_j is close to zero, the term $e^{b_j x_i}$ in the RC-model (A3) or (A5) can be linearized using Taylor approximation, giving

$$\mu_{ij} = R_i C_j e^{b_j x_i} \approx R_i C_j (1 + b_j x_i) \approx y_{i+} y_{+j} (1 + b_j x_i)/y_{++}, \quad \text{(A6)}$$

which can be considered as a one-dimensional equivalent of equation (1) of the main text and is known as the reconstitution formula (Greenacre 1984).The rationale for $y_{i+}y_{+j}/y_{++}$ replacing $R_i C_j$ is that under the null model ($b_j = 0$ for all j) the maximum likelihood (ML) estimate of μ_{ij} is $\hat{\mu}_{ij} = y_{i+}y_{+j}/y_{++}$ when y_{ij} is Poisson distributed. We assume throughout, without loss of generality, that the environmental variable is centred; in particular $\sum_i y_{i+} x_i / y_{++} = 0$. This improves the approximation in (A6).

1.1.2 Transition formulas of CCA approximate ML-equations of constrained RC-model

Whereas equation (A6) is the simplest way to derive the reconstitution formula of (C)CA, it does not yet motivate the transition formulas of CCA (equations (3)-(6) in ter Braak (1986)) . For this, we show that the transition formulas of CCA are an approximation to the ML estimation equations of the Poissonian constrained RC-model.

For more than a single environmental variable, equation (A3) can be extended to a one-dimensional RC-model with p predictors (environmental variables), which in log-linear form is the extension of equation (A4):

$$log(\mu_{ij}) = \tilde{r}_i + \tilde{c}_j + b_j \sum_{l=1}^{p} a_l x_{il}. \quad \text{(A7)}$$

Under the assumption of Poisson distributed abundances $\{y_{ij}\}$, the relevant part of the log-likelihood of the RC-model is

$$l(\theta) = \sum_{i,j} \{ y_{ij}\, log(\mu_{ij}) - \mu_{ij} \}, \quad \text{(A8)}$$

so that, using equation (A7),

$$l(\theta) = \sum_i y_{i+} \tilde{r}_i + \sum_j y_{+j} \tilde{c}_j + \sum_{i,j} y_{ij} b_j \sum_{l=1}^{p} a_l x_{il} - \mu_{++}, \quad \text{(A9)}$$

The ML-equations are obtained by setting the partial derivatives of the log-likelihood l(θ) with respect to each of the parameters to zero. The ML-equations are ,

$$y_{i+} = \mu_{i+}\,,\ y_{+j} = \mu_{+j} \ \ (i = 1, \dots, n; j = 1, \dots, m), \quad \text{(A10)}$$

$$\sum_{i,j} (y_{ij} - \mu_{ij}) b_j x_{il} = 0 \quad (l = 1, \dots, p), \quad \text{(A11)}$$

$$\sum_i (y_{ij} - \mu_{ij}) \tilde{x}_i = 0 \quad (j = 1, \dots, m), \text{ with } \tilde{x}_i = \sum_{l=1}^{p} a_l x_{il}. \quad \text{(A12)}$$

By inserting the approximation

$$\mu_{ij} \approx y_{i+} y_{+j} (1 + b_j \sum_{l=1}^{p} a_l x_{il}) / y_{++}. \quad \text{(A13)}$$

in the ML-equations, we obtain, from equation (A10), that the site and species parameters $\{\tilde{x}_i\}$ and $\{b_j\}$ must both have a weighted mean of zero where the weights are y_{i+} and y_{+j},

respectively. Similarly, we obtain from equation (A12),

$$b_j = (\sum_i y_{ij}\tilde{x}_i/y_{+j})/\sum_i y_{i+}\tilde{x}_i^2/y_{++}. \quad \text{(A14)}$$

and, from equation (A11), after defining,

$$\tilde{x}_i^* = \sum_j y_{ij}b_j/y_{++}, \quad \text{(A15)}$$

$$\mathbf{a} = (\mathbf{X}'\mathbf{R}\mathbf{X})^{-1}\mathbf{X}\mathbf{R}\tilde{\mathbf{x}}^*/(\sum_j y_{+j}b_j^2/y_{++}). \quad \text{(A16)}$$

On recalling that

$$\tilde{x}_i = \sum_{l=1}^{p} a_l x_{il} \text{ or equivalently } \tilde{\mathbf{x}} = \mathbf{X}\mathbf{a}, \quad \text{(A17)}$$

equations (A14) – (A17) are formally equivalent with the transition formulas of CCA (equations (3)-(6) in ter Braak (1986))[1]. This completes our demonstration that the transition formulas of CCA are an approximation to the ML equations of the constrained Poissonian RC-model (A7) under the assumption of closeness to the null model.

1.1.3 Concluding remarks

The transition formulas of CCA are therefore not only an approximation to the ML equations of constrained Gaussian ordination under the assumptions of a species packing model as shown by (ter Braak 1986), but also of the constrained Poissonian RC-model (A7) under the assumption of closeness to the null models. Consequently, CCA can be expected to perform well both close to the null model and far from the null model when the alternative is an unimodal model, in particular, the Gaussian response model.

The assumption that the abundance data are Poisson distributed is of course not very realistic. Nevertheless, the estimators derived from the Poisson are identical to those of the quasi-Poisson (in which the variance is proportional – instead of equal – to the mean, sometimes given the unfortunate name of NB1 (Hardin & Hilbe 2007), and the loss in efficiency for other count distributions (such as the proper negative binomial NB2) might offset the gain in computational efficiency. Because the Poisson distribution is unrealistic for real data, statistical inference proceeds by resampling methods and by permutation methods for statistical significance testing in particular. See ter Braak (2017) for a discussion. ter Braak (2017) also showed that the fitted inertia of a CCA is the Rao score test statistic of the log-linear model if the abundance is Poisson distributed. The Rao score test statistic is asymptotically efficient and is computationally much quicker to compute in this model than the likelihood-ratio test statistic. Computational speed makes the resampling practical in every-day applications.

Finally, note that, if the $\{b_j\}$ would be (known) trait values of species, the $\tilde{x}_i^*$ in equation (A15) is the community weighted mean (CWM) of these values. This result can be phrased otherwise, namely that CCA can be viewed as constructing the best possible (latent) trait of species for a given (set of) environmental variable(s), as CCA optimizes the fourth-

[1] The equivalence can be made complete by substituting u_k for b_j, x_i for $\tilde{x}_i^*$, y_{ik} for y_{ij}, λ for $\sum_i y_{i+}\tilde{x}_i^2/y_{++}$, **b** for **a** and **Z** for **X** and rescaling the species parameters $\{b_j^2\}$ so that $\sum_j y_{+j}b_j^2/y_{++} = 1$.

corner correlation (ter Braak, 2018) and the (WA) site score is a CWM. This result can be phrased otherwise, namely that CCA can be viewed as constructing the best possible (latent) trait of species for a given (set of) environmental variable(s) as CCA optimizes the fourth-corner correlation (ter Braak, Šmilauer & Dray 2018) and the (WA) site score is a CWM.

1.3 References

Goodman, L. A. 1986. Some useful extensions of the usual correspondence analysis approach and the usual log-linear models approach in the analysis of contingency tables. International Statistical Review **54**:243-270.

Greenacre, M. J. 1984. Theory and applications of correspondence analysis. Academic Press, London.

Hardin, J. & Hilbe, J. 2007. Generalized linear models and extensions., 2n ed. Stata Press, College Station, Texas.

Ihm, P., and H. van Groenewoud. 1984. Correspondence analysis and Gaussian ordination. Compstat Lectures **3**:5-60.

ter Braak, C. J. F. 1986. Canonical correspondence analysis: a new eigenvector technique for multivariate direct gradient analysis. Ecology **67**:1167-1179.

ter Braak, C. J. F. 1988. Partial canonical correspondence analysis. Pages 551-558 *in* H. H. Bock, editor. Classification and related methods of data analysis. Elsevier Science Publishers B.V. (North-Holland) http://edepot.wur.nl/241165, Amsterdam.

ter Braak, C.J.F. (2017) Fourth-corner correlation is a score test statistic in a log-linear trait–environment model that is useful in permutation testing. Environmental and Ecological Statistics, 24, 219-242. http://dx.doi.org/10.1007/s10651-017-0368-0

ter Braak, C.J.F., Šmilauer, P. & Dray, S. (2018) Algorithms and biplots for double constrained correspondence analysis. Environmental and Ecological Statistics, 25, 171-197. https://doi.org/10.1007/s10651-017-0395-x

2. Simulation model

Data with n sites, m species and p =12 predictors was simulated using a model with three ordination axes, two of which were constrained; in the first two series n = 30, m = 50, whereas in the third series n = 60, m = 100.

The data simulation started with n draws of three independent sets of variables, each set consisting of four multivariate normal variables each. Each variable had expectation 0 and variance 1. The subsequent variables within each set had a correlation of 0.7. The variables of set 1 are denoted by $\{x_{i1}^*, x_{i2}^*, x_{i3}^*, x_{i4}^*\}$, the variables of set 2 by $\{x_{i5}^*, x_{i6}^*, x_{i7}^*, x_{i8}^*\}$ and the variables of set 3 by $\{x_{i9}^*, x_{i10}^*, x_{i11}^*, x_{i12}^*\}$ (i=1,…, n). The first two sets defined the two constrained ordination axes

$$\tilde{x}_{i1} = a_1 x_{i1}^* + a_2 x_{i2}^* + a_3 x_{i3}^* + a_4 x_{i4}^* \quad \text{(A18)}$$

$$\tilde{x}_{i2} = a_5 x_{i5}^* + a_6 x_{i6}^* + a_7 x_{i7}^* + a_8 x_{i8}^* \quad \text{(A19)}$$

with $a_1 = \cdots = a_8 = 0.3036$, so that their variance is equal to 1. A third unconstrained

axis was formed by an additional independent standard normal variable and is denoted by $\{\tilde{x}_{i3}\}$.

The environmental data were collected in an $n \times 12$ matrix **X** with columns organised in three sets of four variables each

First set: $x_{il} = \sqrt{1-\rho_1^2}x_{il}^* + \rho_1\varepsilon_{ij}$ with $l = 1, \dots, 4$ (A20)

Second set: $x_{il} = \sqrt{1-\rho_2^2}x_{il}^* + \rho_2\tilde{x}_{i1}$ with $l = 5, \dots, 8$ (A21)

Third set: $x_{il} = x_{il}^*$ with $l = 9, \dots, 12$ (A22)

with $\rho_1 = \sqrt{0.1}$, $\rho_2 = 0.7$ and $\varepsilon_{ij} \sim N(0,1)$. Equation (A20) was designed so that the predictors of the first set had unit variance and that 10% of their variance was noise. Similarly, equation (A21) was designed so the predictors of the second set had unit variance and a correlation of ρ_2 with the first axis.

The abundance data was generated by a log-linear model containing the three ordination axes. In the first series the model was

$$log(\mu_{ij}) = \tilde{a}_0 + \tilde{r}_i + \tilde{c}_j + \tilde{b}_0(\tilde{b}_{j1}\tilde{x}_{i1} + 0.5\tilde{b}_{j2}\tilde{x}_{i2}) + \tilde{b}_{j3}\tilde{x}_{i3} \quad \text{(A23)}$$

with $\tilde{a}_0 = \log(10)$, $\tilde{r}_i \sim N(0,\sigma_1^2)$, $\tilde{c}_j \sim N(0, 0.25)$, $\tilde{b}_{j1} \sim N(0,1)$, $\tilde{b}_{j2} \sim N(0,1)$, so that, with $\tilde{b}_0 \neq 0$, the first axis is four times as important in terms of variance than the second. The parameter $\tilde{b}_0$ in equation (A23) is the (overall) effect size of the constrained axes on species abundance. Parameter $\tilde{b}_{j3}$ is the size of the rank 1 noise and σ_1 is the standard deviation of the site log-linear main effects with values 0 or 0.5.

In the second series, which is on testing dimensionality, the term $\tilde{b}_0\tilde{b}_{j1}\tilde{x}_{i1}$ was replaced by a quadratic term

$$-\left(\tilde{x}_{i1} - u_{j1}\right)^2/2t_j^2 \quad \text{(A24)}$$

with $u_{j1} \sim N(0,2)$ and t_j is exponentially distributed with mean 1. Also, the term $\tilde{b}_0\tilde{b}_{j2}\tilde{x}_{i2}$ was replaced by the parameter $\tilde{b}_2\tilde{b}_{j1}\tilde{x}_{i2}$ so that $\tilde{b}_2$ is the effect of the second axis.

The model in the third series, which is on the (in)sensitivity of CCA to detect environmental main effects, was like the model and parameters of the first series except that n=60 and m = 100 and

$$\tilde{r}_i = \sigma_1(\rho_3 x_{i1} + \sqrt{1-\rho_3^2}\varepsilon_i) \text{ with } l = 1, \dots, 4 \quad \text{(A25)}$$

with $\varepsilon_i \sim N(0,1)$, so that the site log-linear main effect had a standard deviation of σ_1 and a correlation of ρ_3 with the first environmental variable. The value of σ_1 was set to 0.5 so as to obtain a large environmental effect when ρ_3 is moderate to large.

Species abundance was drawn from a negative binomial distribution with mean μ_{ij}. The variance was $\mu_{ij} + \varphi\mu_{ij}^2$ where ϕ is the overdispersion compared to the Poisson distribution.

A summary of the main parameters of the models is given in Table A1.

Table A1. The main parameters of the models and the series and figures of the main text in which they appear. Figure numbers between brackets had a fixed value of the parameter of the corresponding row; the fixed value is given in the column Parameter.

Series	Figures	Parameter	Meaning	Abbreviation
1	1	$\tilde{b}_0$	Overall effect size	Effect size
1	1 (2,3,4)	σ_1 0.5	Standard deviation of the site log-linear main effects $\{\tilde{r}_i\}$	site total sd
1,2,3	1 (2,3,4)	ϕ 0.2	overdispersion of the count compared to that in the Poisson distribution	overdispersion
1,2,3	1,2,4 3	$\tilde{b}_{j3}$ 0.5	Effect of the third axis that was independent of the environmental variables in the data (unconstrained axis, structured noise)	size rank 1 noise
2	3	$\tilde{b}_2$	Effect size of the second axis	Effect size of dimension 2
3	4	ρ_3	Correlation of the site log-linear main effects with the first environmental variable $\{x_{i1}\}$	ρ(site main effect, x_1)

3. Simulation results for skewed and binary predictors

This appendix uses the same simulated data as the main text, except that the predictor data are either exponentiated, *i.e.* $x_{ij} \leftarrow \exp(x_{ij})$, so that the predictors are skewed as concentrations of chemicals typically are, or made binary (1 if $x_{ij} > 0$ else 0). The results are reported as Figures A1 – A8 which match with Figures 1-4 in the main text, with Figures A1-A4 for the exponentiated predictor data and Figure A5-A8 for the binarized predictors.

1.1.4 Skewed predictors

Figure A1

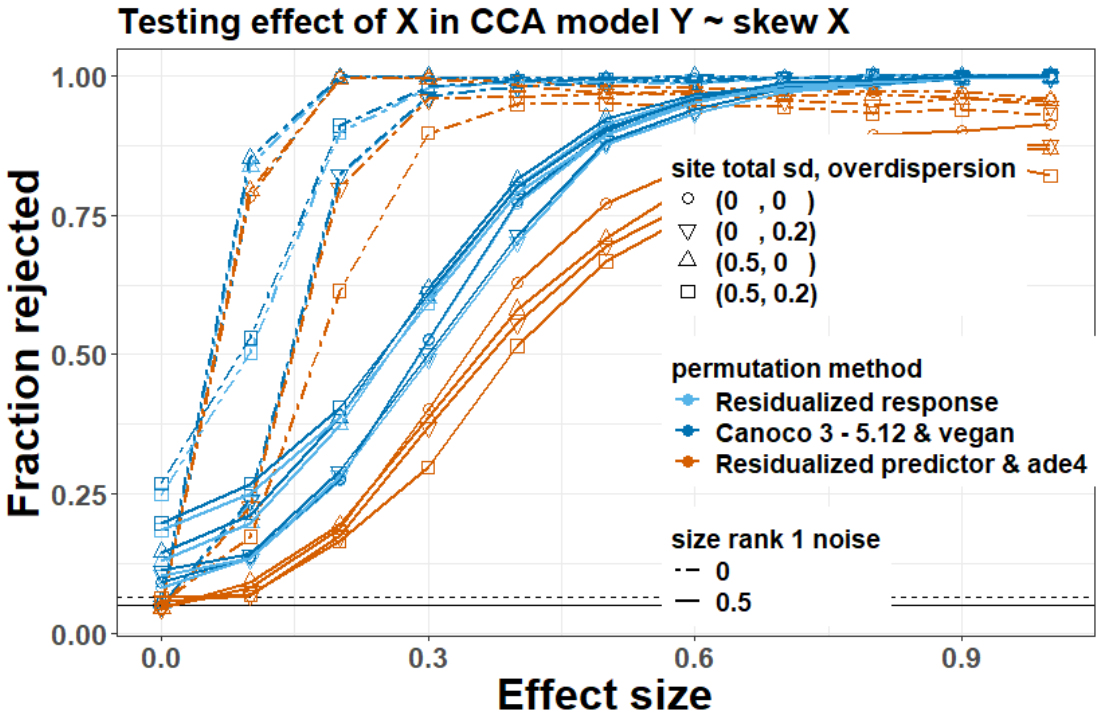

Figure A1. Influence of noise types on the rejection rates (Type I error rate if effect size = 0, power otherwise) of three permutation methods for testing the effect of twelve predictors (**X**) on abundance data (**Y**) using CCA with the model **Y**~**X** (n=30, m=50, p=12) (data generated using the loglinear simulation model; Effect size = overall effect size; noise types: (1) site total sd = standard deviation of the site main effect, (2) overdispersion = overdispersion parameter of the negative binomial (0 = Poissonian), (3) size rank 1 noise = size of the effect of an unobserved predictor that is independent of **X**). The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A2

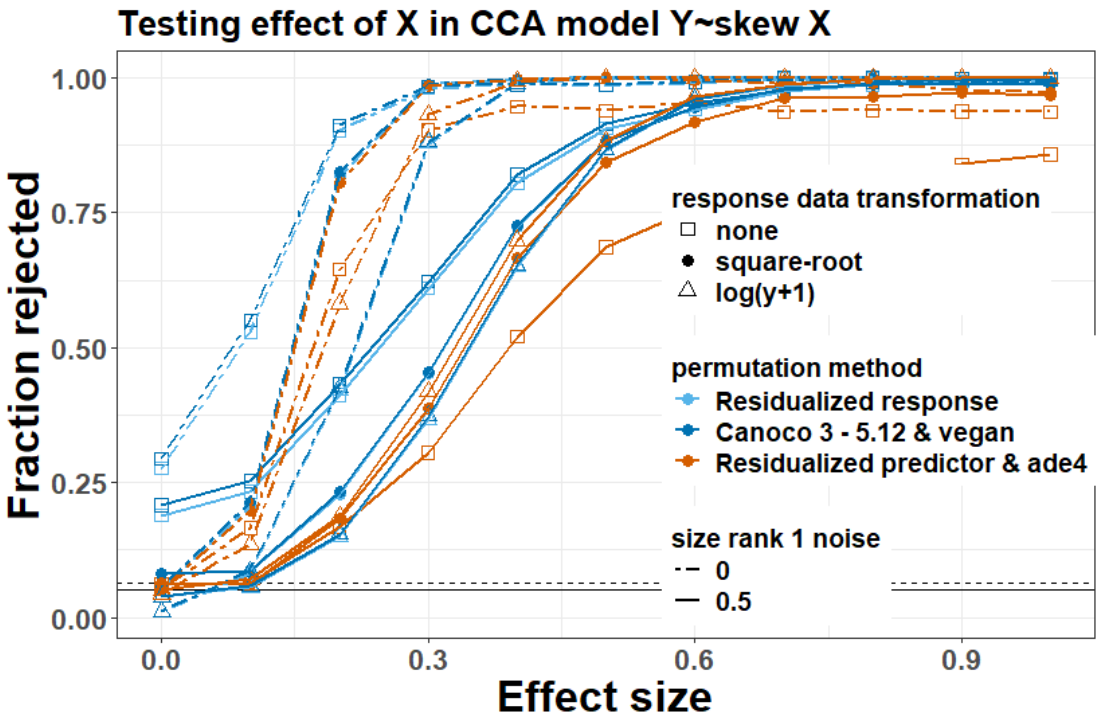

Figure A2. The influence of data transformation and noise on the rejection rates (Type I error rate if effect size = 0, power otherwise) of three permutation methods for testing the effect of twelve predictors (**X**) on transformed abundance data with CCA using model **Y**~**X** (n=30, m=50, p=12). Data generated using the loglinear simulation model with overdispersion 0.2 and a standard deviation of 0.5 of the site main effect. For effect size and size rank 1 noise, see legend Figure 1. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A3

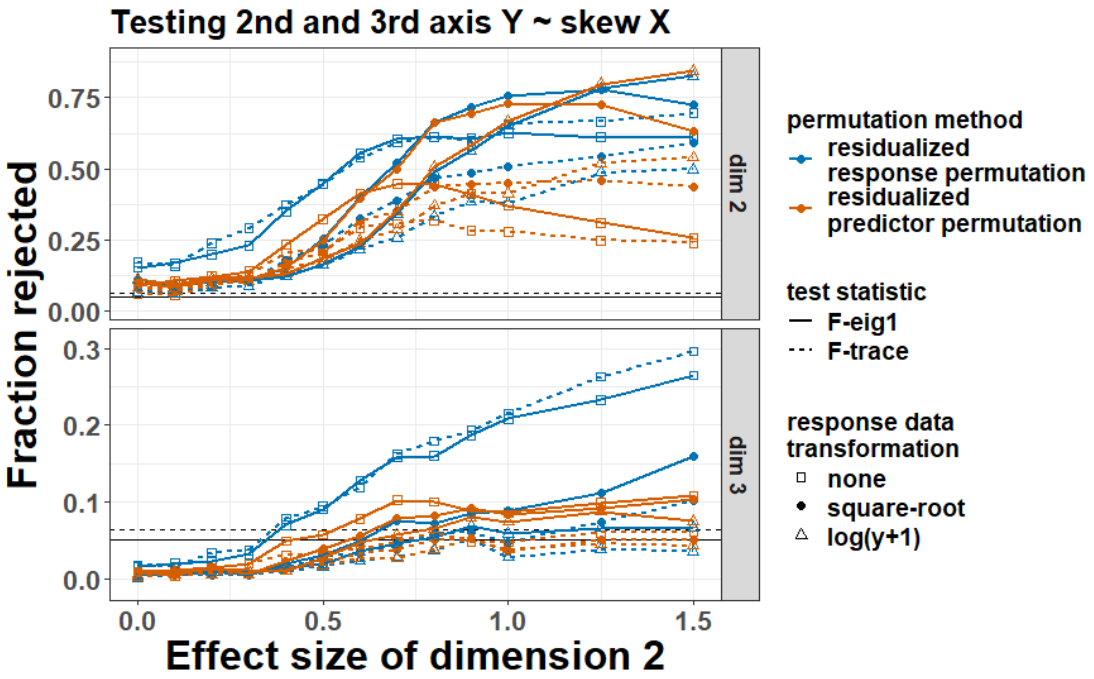

Figure A3. Rejection rates of testing the second and third axes against the effect size of the second axis by CCA using two alternative test statistics (F_{eig} and F_{trace}) with n=30, m=50, p=12. The rejection rate for the first axis, which had Gaussian response in this simulations series, was close to 1 everywhere. Data generated using overdispersion 0.2, a standard deviation of 0.5 of the site main effect and rank 1 noise of 0.5. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A4

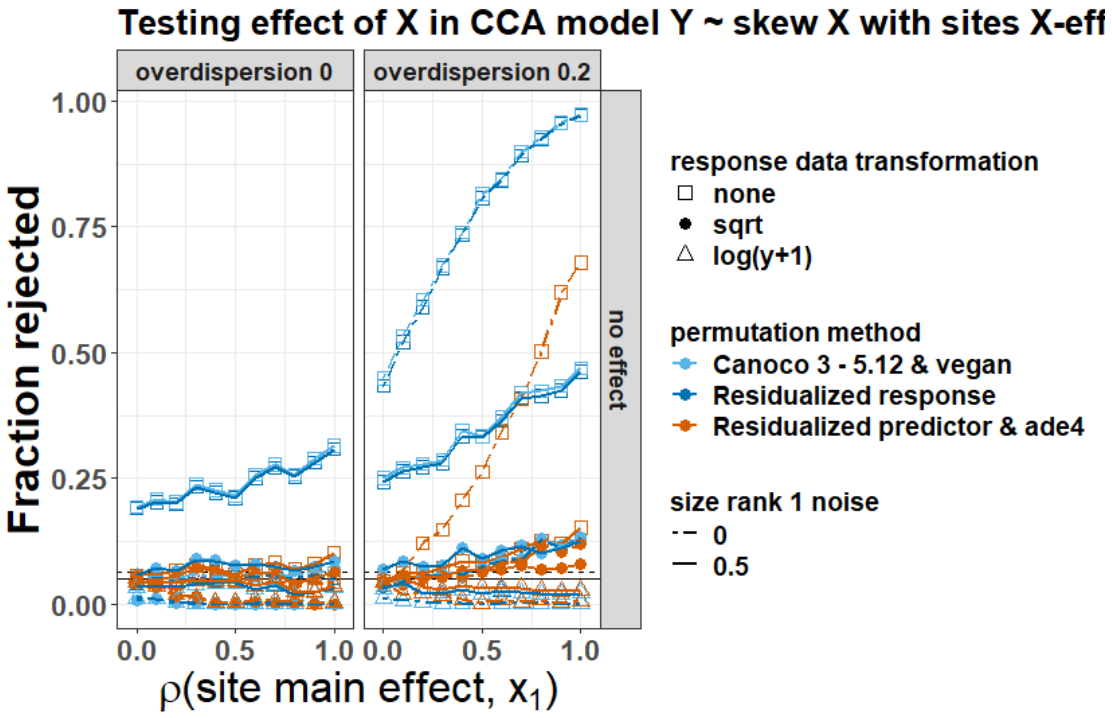

Figure A4. Type I error rate of testing the effect of twelve predictors (**X**) on transformed abundance data with CCA using model **Y**~**X** (n=60, m=100, p=12) in relation to the correlation of one of the predictors (x_1) with the log-linear site main effect, with the influence of data transformation, overdispersion and rank 1 noise. Data generated as in Figure 1 with effect size = 0, except that the site main effects were made correlated with x_1. The standard deviation of the site main effects was 0.5. For size rank 1 noise, see legend Figure 1. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

1.1.5 Binary predictors

Figure A5

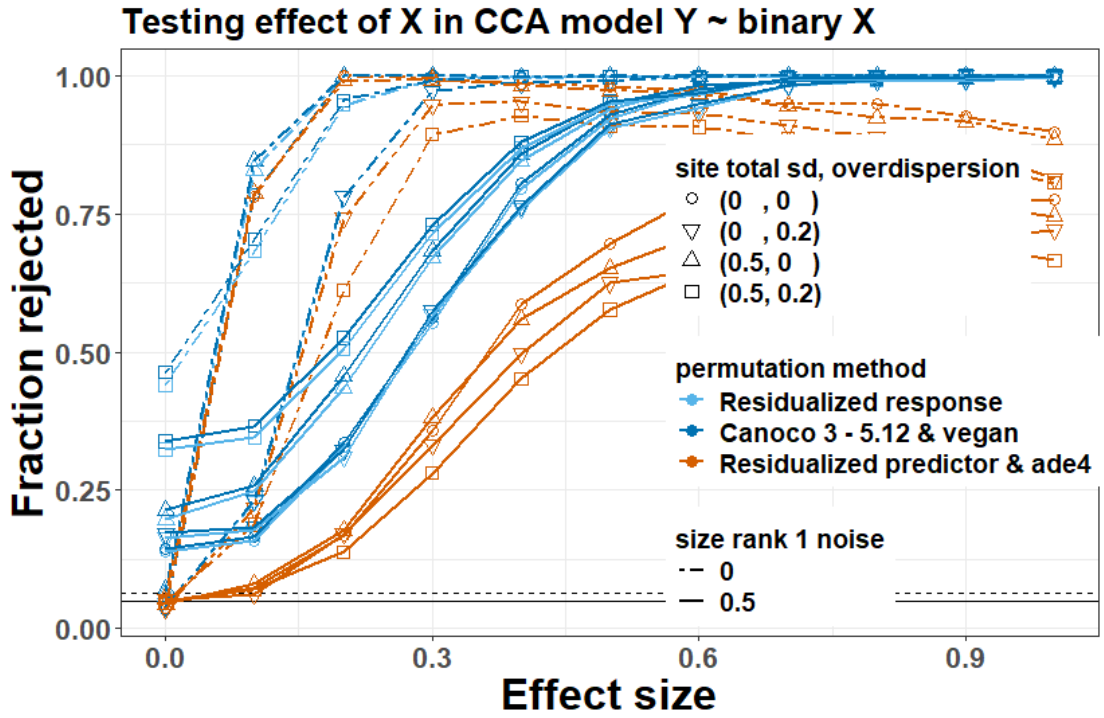

Figure A5. Influence of noise types on the rejection rates (Type I error rate if effect size = 0, power otherwise) of three permutation methods for testing the effect of twelve binary predictors (**X**) on abundance data (**Y**) using CCA with the model **Y**~**X** (n=30, m=50, p=12) (data generated using the loglinear simulation model; Effect size = overall effect size; noise types: (1) site total sd = standard deviation of the site main effect, (2) overdispersion = overdispersion parameter of the negative binomial (0 = Poissonian), (3) size rank 1 noise = size of the effect of an unobserved predictor that is independent of **X**). The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A6

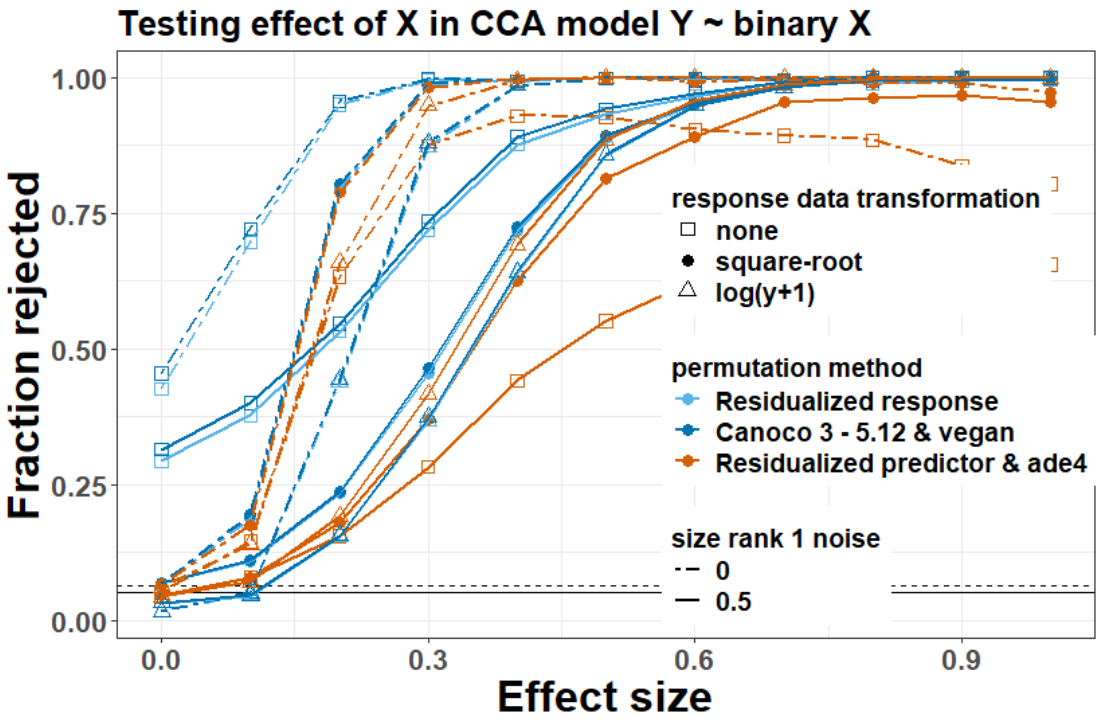

Figure A6. The influence of data transformation and noise on the rejection rates (Type I error rate if effect size = 0, power otherwise) of three permutation methods for testing the effect of twelve binary predictors (**X**) on transformed abundance data with CCA using model **Y**~**X** (n=30, m=50, p=12). Data generated using the loglinear simulation model with overdispersion 0.2 and a standard deviation of 0.5 of the site main effect. For effect size and size rank 1 noise, see legend Figure 1. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A7

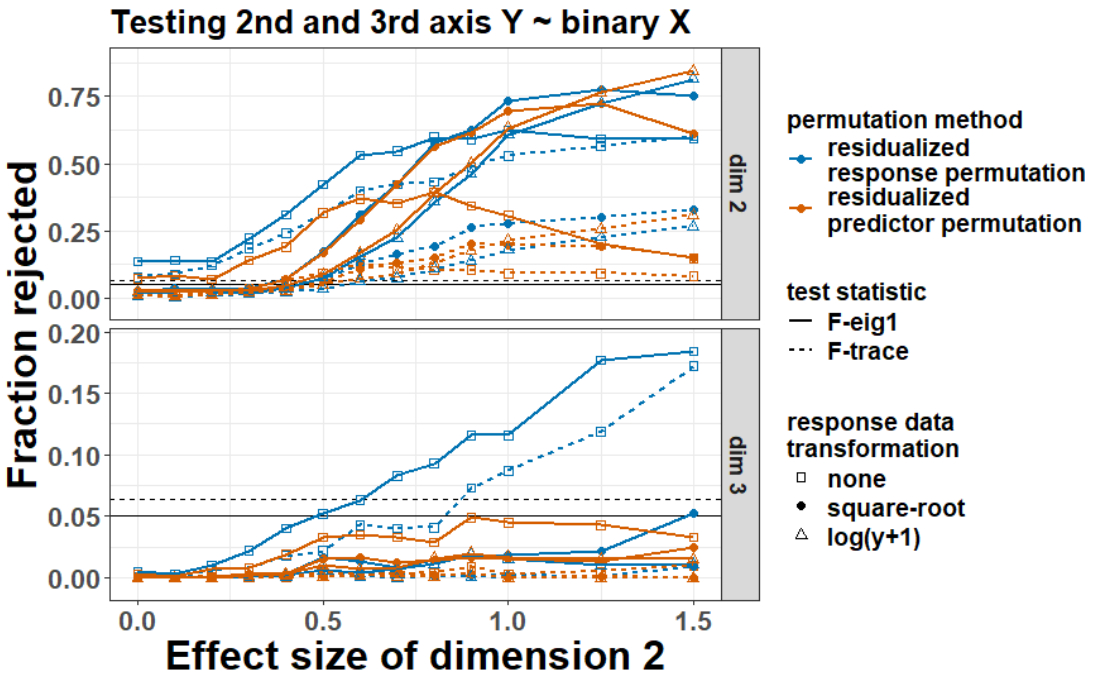

Figure A7. Rejection rates of testing the second and third axes against the effect size of the second axis by CCA using two alternative test statistics (F_{eig} and F_{trace}) with n=30, m=50, p=12. The rejection rate for the first axis, which had Gaussian response in this simulations series, was close to 1 everywhere. Data generated using overdispersion 0.2, a standard deviation of 0.5 of the site main effect and rank 1 noise of 0.5. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

Figure A8

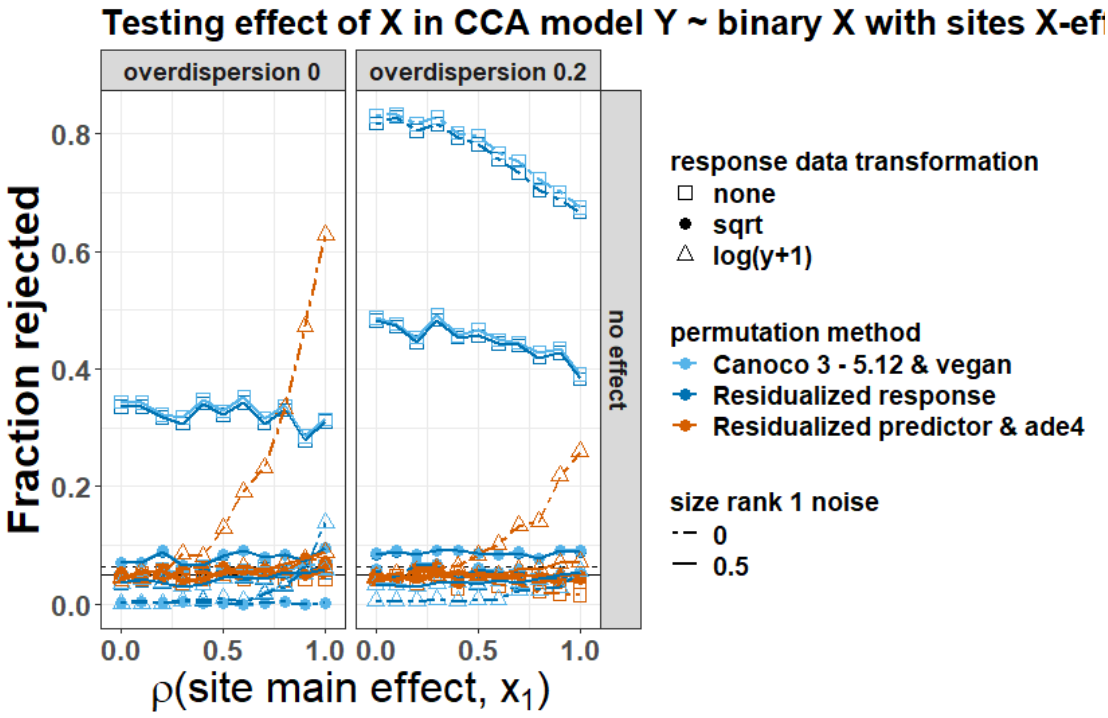

Figure A8. Type I error rate of testing the effect of twelve binary predictors (**X**) on transformed abundance data with CCA using model **Y**~**X** (n=60, m=100, p=12) in relation to the correlation of one of the predictors (x_1) with the log-linear site main effect, with the influence of data transformation, overdispersion and rank 1 noise. Data generated as in Figure 1 with effect size = 0, except that the site main effects were made correlated with x_1. The standard deviation of the site main effects was 0.5. For size rank 1 noise, see legend Figure 1. The horizontal solid line is at the nominal significance threshold; rates (from 1000 simulations) above the dashed line (at 0.064) are significantly greater than 0.05.

4. First case study: geographic trend in neotropical forests before and after adjustment for climate

```
rm(list=ls(all=TRUE))  # remove all existing items from the workspace

source("Rfunctions/rXY_permutation.r")
source("rfunctions/Test_CCA_functions.r")
#mysvd <- svd if eig1 test is needed
mysvd<- dummysvd # if no eig1 tests are needed (avoids svd)


nrepet = 199
n_simul <- 1000

nominal.level <- 0.05
with_vegan_ade4 <- FALSE
set.seed(1357)


# Read the data ----------------------------------------------------------
----

Y <- read.csv("data/Pinho_SumCountsClustersAnon.csv")
Y <- Y[,-c(1,2)]
#Y[5,] <- Y[5,]/3 # the largest cluster
dim(Y)

## [1]   59 3416

env0 <- read.csv("data/Pinho_MeanEnvironmentClusters.csv")
names(env0)

##  [1] "cluster"              "Biogeographic.region"
##  [3] "Latitude"             "Longitude"
##  [5] "Degrees2equator"      "Dist2Ocean"
##  [7] "MAT"                  "TS"
##  [9] "PET"                  "MAP"
## [11] "PS"

dim(env0)
```

```
## [1] 59 11

# Hugely different site totals ----------------------------------------
----

range(rowSums(Y)) # 35 - 36915

## [1]    35 36898

# Select the variables ------------------------------------------------
----

X<- env0[, c("Longitude"),drop = FALSE]
Z <- env0[,c("MAT","TS","PET","MAP","PS")]




# Simple and conditional effect of Longitude --------------------------
---------------------------
set.seed(1235)
P_simple <- test_vegan_ade4_randperm_CCA_transf(Y,X, Z = NULL, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 )
P_conditional_on_climate <-  test_vegan_ade4_randperm_CCA_transf(Y,X,Z,
nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 )
P_simple

##         CCA_Y CCA_Ycan3 CCA_X
## pow_1   0.005     0.005 0.010
## pow_0.5 0.005     0.005 0.005
## pow_0   0.005     0.005 0.005

P_conditional_on_climate

##         CCA_Y CCA_Ycan3 CCA_X
## pow_1   0.005     0.005 0.400
## pow_0.5 0.005     0.005 0.065
## pow_0   0.005     0.005 0.045



# Simple and conditional effect of a completely random normal variable
------------------------------------------------------

if (with_vegan_ade4) Ntest <-5 else Ntest <- 3
pval_mat_simple <- array(NA, dim =c(n_simul,3,Ntest))
pval_mat_conditional <- array(NA, dim =c(n_simul,3,Ntest))
nran <- n_simul * prod(dim(X))

Xrandom_array <-array(rnorm(nran), dim=c(n_simul, dim(X)))

for (i in 1:n_simul){
```

```
  Xran <- data.frame(Xran = Xrandom_array[i,,])
  pval_mat_simple[i,,]<- test_vegan_ade4_randperm_CCA_transf(Y,Xran,Z =
NULL, nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4, seed = i)
  pval_mat_conditional[i,,]<-
test_vegan_ade4_randperm_CCA_transf(Y,Xran,Z, nrepet = nrepet,
with_vegan_ade4 =with_vegan_ade4, seed = i)

}
# fraction of rejections
Fraction_rejected_simple <- colMeans(pval_mat_simple <= nominal.level,
na.rm = TRUE)
Fraction_rejected_conditional <- colMeans(pval_mat_conditional <=
nominal.level, na.rm = TRUE)

dimnames(Fraction_rejected_conditional) <-
dimnames(Fraction_rejected_simple) <-
dimnames(P_conditional_on_climate)
Fraction_rejected_simple

##         CCA_Y CCA_Ycan3 CCA_X
## pow_1   0.888     0.905 0.059
## pow_0.5 0.463     0.505 0.048
## pow_0   0.330     0.382 0.048

Fraction_rejected_conditional

##         CCA_Y CCA_Ycan3 CCA_X
## pow_1   0.830     0.863 0.058
## pow_0.5 0.316     0.366 0.046
## pow_0   0.226     0.262 0.043
```

See Appendix A5 for meaning of the abbreviations in the output.

5. Case studies using data from R-packages

P-values of hypotheses tested on data in R packages ade4, vegan and mvabund by CCA using

CCA_Y: residualized response permutation

CCA_Ycan3: residualized response permutation ignoring the intercept as used in Canoco from version 3 to 5.12

vegan: permutation using vegan 2.5-7

ade4: permutation using ade4 1.7-15

CCA_X: residualized predictor permutation

Three type of response data transformation were used:

pow_1 : no transformation

pow_0.5 : square-root transformation

pow_0 : log(y+1) transformation

The R-script is available in the online repository (file real_data_comparison.r). The log of running the script, using markdown, is as follows:

Real_data_comparison.r

```
rm(list=ls(all=TRUE))  # remove all existing items from the workspace

source("Rfunctions/rXY_permutation.r")
source("rfunctions/Test_CCA_functions.r")
#mysvd <- svd if eig1 test is needed
mysvd<- dummysvd # if no eig1 tests are needed (avoids svd)

library(ade4)
library(vegan)

nrepet = 1999
with_vegan_ade4 <- TRUE
#with_vegan_ade4 <- FALSE
Pvals <- list()


# data from ade4  ------------------------------------------------------
---------

data("doubs", package = "ade4")

(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(doubs$fish,doubs$env, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

data("dunedata", package = "ade4" )
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(dunedata$veg,dunedata$envir, nrepet
= nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##          CCA_Y CCA_Ycan3  vegan  ade4 CCA_X
## pow_1   0.0020    0.0020 0.0020 0.001 0.001
## pow_0.5 0.0015    0.0015 0.0015 0.001 0.001
## pow_0   0.0015    0.0015 0.0015 0.001 0.001

# data from vegan  ----------------------------------------------------
----------

data("mite", "mite.env","mite.xy", package = "vegan")
# test on effect of env on the species data
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(mite,mite.env, nrepet = nrepet,
with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

names(mite.xy)[2]<-"yy" # y-> yy to avoid name conflict in
ade4::randtest(cca1,...)
# test on geographic trend in the species data
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(mite,mite.xy, nrepet = nrepet,
with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

# effect of env conditional on linear geography
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(mite,mite.env, Zdf = mite.xy,
nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04   NA 5e-04
## pow_0.5 5e-04     5e-04 5e-04   NA 5e-04
## pow_0   5e-04     5e-04 5e-04   NA 5e-04

data("BCI", package = "vegan")
data("BCI.env", package = "vegan")
# set constant or near constant variables to null
BCI.env$Geology <- BCI.env$Age.cat <- BCI.env$Precipitation <-
BCI.env$Elevation <- NULL
# effect of env on the species data?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(BCI,BCI.env[, -c(1,2)], nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
```

```
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

#  geography trend (linear) in species data?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(BCI,BCI.env[, c(1,2)], nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

# effect of env conditional on linear geographic trend?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(BCI,BCI.env[, -c(1,2)],BCI.env[,
c(1,2)], nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04   NA 5e-04
## pow_0.5 5e-04     5e-04 5e-04   NA 5e-04
## pow_0   5e-04     5e-04 5e-04   NA 5e-04

data("sipoo", package = "vegan")
data("sipoo.map", package = "vegan")
names(sipoo.map)

## [1] "N"    "E"    "area"

# effect of area
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(sipoo,sipoo.map$area, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   0.001     5e-04 5e-04 5e-04 5e-04
## pow_0.5 0.001     5e-04 5e-04 5e-04 5e-04
## pow_0   0.001     5e-04 5e-04 5e-04 5e-04

# geographic trend?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(sipoo,sipoo.map[, c(1,2)], nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan   ade4  CCA_X
## pow_1   0.148     0.121 0.121 0.5155 0.5155
## pow_0.5 0.148     0.121 0.121 0.5155 0.5155
## pow_0   0.148     0.121 0.121 0.5155 0.5155

# effect  of area conditional on linear geography trend
(Pvals[[length(Pvals)+1]] <-
```

```
test_vegan_ade4_randperm_CCA_transf(sipoo,sipoo.map$area,sipoo.map[,
c(1,2)], nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan ade4 CCA_X
## pow_1   0.002     0.002 0.002   NA 0.001
## pow_0.5 0.002     0.002 0.002   NA 0.001
## pow_0   0.002     0.002 0.002   NA 0.001

data("varespec", package = "vegan")
data("varechem", package = "vegan")
# too many predictor variables for this small dataset?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(varespec,varechem, nrepet = nrepet,
with_vegan_ade4 =with_vegan_ade4 ))

##          CCA_Y CCA_Ycan3  vegan   ade4  CCA_X
## pow_1   0.0320    0.0320 0.0320 0.0365 0.0365
## pow_0.5 0.0635    0.0630 0.0630 0.0785 0.0785
## pow_0   0.0720    0.0715 0.0715 0.1015 0.1015

# data from mvabund ----------------------------------------------------
-----------------

data("spider", package = "mvabund")
colnames(spider$x)

## [1] "soil.dry"      "bare.sand"     "fallen.leaves" "moss"
## [5] "herb.layer"    "reflection"

(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(spider$abund,spider$x, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

data("tikus", package = "mvabund")
#View(tikus$abund)
str(tikus$x)

## 'data.frame':    60 obs. of  2 variables:
##  $ time: Factor w/ 6 levels "81","83","84",..: 1 1 1 1 1 1 1 1 1 1
...
##  $ rep : Factor w/ 10 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9
10 ...

(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(tikus$abund,tikus$x, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04
```

```
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(tikus$abund,tikus$x$time, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04
```

```
# location effect ignoring time
# interest?
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(tikus$abund,tikus$x$rep, nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan   ade4  CCA_X
## pow_1   5e-04     5e-04 5e-04 0.0005 0.0005
## pow_0.5 5e-04     5e-04 5e-04 0.0010 0.0010
## pow_0   5e-04     5e-04 5e-04 0.0025 0.0025
```

```
# time given location (rep)
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(tikus$abund,tikus$x$time,Z =
tikus$x$rep, nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04   NA 5e-04
## pow_0.5 5e-04     5e-04 5e-04   NA 5e-04
## pow_0   5e-04     5e-04 5e-04   NA 5e-04
```

```
# location given time (rep)
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(tikus$abund,tikus$x$rep, Z =
tikus$x$time, nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04   NA 5e-04
## pow_0.5 5e-04     5e-04 5e-04   NA 5e-04
## pow_0   5e-04     5e-04 5e-04   NA 5e-04
```

```
data("antTraits", package = "mvabund")
names(antTraits)
```

```
## [1] "abund"  "env"    "traits"
```

```
#str(antTraits$env)
#str(antTraits$abund)
```

```r
#str(antTraits$traits)
# effect of effect on the abundance
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(antTraits$abund,antTraits$env,nrepe
t = nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04
```

```r
# no effect of traits on the abundance detected
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(t(antTraits$abund),antTraits$traits
,nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##          CCA_Y CCA_Ycan3 vegan   ade4  CCA_X
## pow_1   0.4680     0.433 0.433 0.7675 0.7675
## pow_0.5 0.6940     0.662 0.662 0.2960 0.2960
## pow_0   0.6385     0.603 0.603 0.2130 0.2130
```

```r
data("solberg", package = "mvabund")
names(solberg)
```

```
## [1] "abund" "x"
```

```r
# almost constant site totals (98-102), very variable species totals
(1-189)
# untransformed: no effect detectable; after sqrt or log: effect
detectable
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(solberg$abund,solberg$x,nrepet =
nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##          CCA_Y CCA_Ycan3  vegan   ade4  CCA_X
## pow_1   0.1840    0.1840 0.1840 0.1770 0.1770
## pow_0.5 0.0295    0.0285 0.0285 0.0300 0.0300
## pow_0   0.0180    0.0180 0.0180 0.0175 0.0175
```

```r
data("Tasmania", package = "mvabund")
names(Tasmania)
```

```
## [1] "abund"     "copepods"  "nematodes" "treatment" "block"
## [6] "tr.block"
```

```r
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(Tasmania$abund,Tasmania$treatment,Z
= Tasmania$block, nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))
```

```
##          CCA_Y CCA_Ycan3  vegan ade4 CCA_X
## pow_1   0.0015    0.0015 0.0015   NA 1e-03
## pow_0.5 0.0010    0.0005 0.0005   NA 5e-04
```

```
## pow_0   0.0005    0.0005 0.0005   NA 5e-04

# data from TraitEnvMLMWA ------------------------------------------
-----

#install.packages("remotes")
#remotes::install_github("CajoterBraak/TraitEnvMLMWA")
data("Revisit", package = "TraitEnvMLMWA")
#str(Revisit)
#str(Revisit$y)
Y <- matrix(Revisit$y[,1], nrow = 52, ncol =75)
X <- Revisit$env[1:52]
T1 <-matrix(Revisit$trait, nrow = 52, ncol =75)[1,]
# a strong  environment gradient topographic moisture gradient
(Pvals[[length(Pvals)+1]] <- test_vegan_ade4_randperm_CCA_transf(Y,X,
nrepet = nrepet, with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan  ade4 CCA_X
## pow_1   5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0.5 5e-04     5e-04 5e-04 5e-04 5e-04
## pow_0   5e-04     5e-04 5e-04 5e-04 5e-04

# a weak trait response, stronger with sqrt and log-transformation in
predictor permutation,
# for the log perhaps helped by the trait main effect; see
https://doi.org/10.1111/2041-210X.13278
(Pvals[[length(Pvals)+1]] <-
test_vegan_ade4_randperm_CCA_transf(t(Y),X= T1, nrepet = nrepet,
with_vegan_ade4 =with_vegan_ade4 ))

##         CCA_Y CCA_Ycan3 vegan   ade4  CCA_X
## pow_1   5e-04     5e-04 5e-04 0.0420 0.0420
## pow_0.5 5e-04     5e-04 5e-04 0.0085 0.0085
## pow_0   5e-04     5e-04 5e-04 0.0045 0.0045

length(Pvals)

## [1] 24

#save.image("real_data.rdata")
```