

Todo list

Describe experiment	21
-------------------------------	----

EDUARD SZÖCS

QUANTITATIVE ECOTOXICOLOGY

WITH R!

This document was created using \LaTeX , knitr and the tufte book class.

Copyright © 2013 Eduard Szöcs



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License](#).

First draft, October 2013

Contents

1	<i>Introduction</i>	9
2	<i>The Measurement Process</i>	11
2.1	<i>Winsorized Mean and Standard Deviation</i>	11
2.2	<i>Probability Plotting</i>	12
3	<i>Bioaccumulation</i>	13
4	<i>Tests for Detection of Chronic Lethal and Sublethal Stress</i>	15
5	<i>Lethal and Other Quantal Responses to Stress</i>	17
5.1	<i>Fitting dose-response models</i>	17
6	<i>Population and Metapopulation Effects</i>	19
7	<i>Community Effects</i>	21
7.1	<i>Species Richness</i>	21
7.2	<i>Analyzing mesocosm data</i>	21
7.3	<i>Species Sensitivity Distributions</i>	27
	<i>R Session Info</i>	29

<i>Bibliography</i>	31
---------------------	----

List of Figures

2.1	A histogramm of the so ₄ data.	11
5.1	LD ₅₀ . Source: http://xkcd.com/1260/	17
7.1	Principal response curves (PRC) of pyrifos data	23
7.2	Responses of <i>G. pulex</i> and <i>C. horaria</i> to chlorpyrifos.	23
7.3	PRC - Explained variance	25

List of Tables

7.1	3 ditches observed for 4 weeks and a possible permutation.	25
-----	--	----

1

Introduction

```
require(devtools)  
install_github("qetx", "EDiLD")
```

```
require(qetx)
```


2

The Measurement Process

2.1 Winsorized Mean and Standard Deviation

The following sulfate concentrations (mg/L) were measured during a routine water quality survey of the Savannah River (South Carolina). The data is available in the `qetx` package ¹:

```
data(so4)
```

```
so4
## [1] 1.3 2.3 2.6 3.3 3.5 3.5 3.6 4.0 4.1 4.5 5.2 5.6
## [13] 5.7 6.1 6.2 6.5 6.9 7.1 7.7 7.9 9.9

length(so4)
## [1] 21

mean(so4)
## [1] 5.119

sd(so4)
## [1] 2.137
```

So there are 21 measurements with a mean of 5.12 mg/L and a standard deviation of 2.14 mg/L.

Suppose we have a detection limit of 2.5 mg/L and want to win-sorize values below LOD, i.e. replace the two lowest values by 2.6 mg/L and the two highest values by 7.7 mg/L.

¹ Note that in this case you do not have to assign the data to a name.

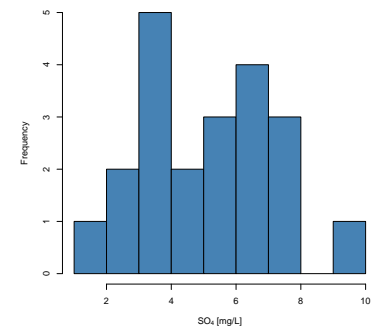


Figure 2.1: A histogram of the `so4` data.

Happily there is function in the `qetx` to do this for us: `winsor()`. This function takes a vector of values and a second argument specifying how many values should be winsorized (either by giving a LOD-value or the number of values on each side) ².

```
so4_w <- winsor(so4, lod = 2.5)
so4_w

## [1] 2.6 2.6 2.6 3.3 3.5 3.5 3.6 4.0 4.1 4.5 5.2 5.6
## [13] 5.7 6.1 6.2 6.9 6.5 7.1 7.7 7.7 7.7
## attr("width")
## [1] 2
```

² Look at the source of this function - type the function name into the console - to see which computations are done.

This give the expected results, moreover we see that on each end two observations were modified ³.

```
mean(so4_w)

## [1] 5.081

sd(so4_w)

## [1] 1.792

sd_winsor(so4_w)

## [1] 2.24
```

³ stored within the attribute 'width' of the resulting vector. **TODO: verbatim within sidenote.**

The Winsorized mean (\bar{x}_w) now is 5.08 mg/L, the standard deviation of the modified data set (s) is 1.79 mg/L and the Winsorized standard deviation (s_w) 2.24 mg/L.

2.2 Probability Plotting

3

Bioaccumulation

4

Tests for Detection of Chronic Lethal and Sub-lethal Stress

5

Lethal and Other Quantal Responses to Stress

5.1 Fitting dose-response models

```
data(salt)
head(salt)
```

##	dead	total	conc
## 1	16	76	10.3
## 2	22	79	10.8
## 3	40	77	11.6
## 4	69	76	13.2
## 5	78	78	15.8
## 6	77	77	20.1



THE LD₅₀ OF TOXICITY DATA IS
2 KILOGRAMS PER KILOGRAM.

Figure 5.1: LD₅₀. Source: <http://xkcd.com/1260/>

6

Population and Metapopulation Effects

7

Community Effects

7.1 Species Richness

```
data(abu)
abu

##      Sp1 Sp2 Sp3 Sp4 Sp5 Sp6 Sp7 Sp8 Sp9 Sp10
## -1     12  11  10   8   8   5   2   2   1   0
## -0.5    12  12  10   8   8   4   3   2   2   1
##  0      58  21   3   2   1   0   0   0   0   0
##  0.5     18  16  15   5   4   0   0   0   0   0
##  1       11  11  11   8   7   3   2   1   1   0
##  1.7      8  12  13   3   3   3   1   2   0   0
##  2.7     12  10   8  11   8   4   1   1   2   0
##  5.3     10  11   8   6   6   5   4   2   1   1
```

7.2 Analyzing mesocosm data

Example data

Here we will analyze the pyrifos data set from [Van den Brink and Ter Braak \(1999\)](#) which is shipped with the vegan package.

Describe experiment

```
require(vegan)
data(pyrifos)
head(pyrifos[, c(1:8)])

##      Simve Daplo Cerpu Alogu Aloco Alore Aloaf Copsp
## w.4.c1 3.951    0    0    0    0    0    0 2.773
## w.4.c2 2.303    0    0    0    0    0    0 2.079
## w.4.c3 4.595    0    0    0    0    0    0 3.761
## w.4.c4 2.398    0    0    0    0    0    0 3.296
```

```
## w.4.c5 4.025      0      0      0      0      0      0 3.466
## w.4.c6 2.303      0      0      0      0      0      0 2.197
```

So rows correspond to samples and columns are the species (with abbreviated names), a usual species x sites matrix. The column names code treatment and time, but we will create a separate data.frame with information about experimental ditch, sampling time and treatment:

```
ditch <- gl(12, 1, length = 132)
week <- gl(11, 12, labels = c(-4, -1, 0.1, 1, 2, 4, 8, 12,
  15, 19, 24))
dose <- factor(rep(c(0.1, 0, 0, 0.9, 0, 44, 6, 0.1, 44,
  0.9, 0, 6), 11))
pyrifos_env <- data.frame(ditch, week, dose)
```

Introduction

Principle Response Curves (PRC)¹ are commonly used for analyzing ecotoxicological mesocosm experiments. PRC analyses the change of a community due to a treatment over time and is a special form of Redundancy Analysis (RDA)².

¹ Van den Brink and Ter Braak, 1999

² Legendre and Legendre, 2013

Overall pattern

With this a hand we can easily calculate and plot (Figure 7.1)³ the PRC using the prc() function:

```
pyrifos_prc <- prc(response = pyrifos, treatment = dose,
  time = week)
pyrifos_prc_sum <- summary(pyrifos_prc, scaling = 1)
```

```
plot(pyrifos_prc, select = abs(pyrifos_prc_sum$sp) > 1,
  scaling = 1)
```

³ Note that only species with scores greater or smaller than 1 are displayed to avoid cluttering of the plot

The plot shows on the x axis the time and on the y-axis the difference from the control treatments. The farther apart from the x-axis the more different are the communities compared to the control (you can say the x axis represents the control).

We see a clearly treatment-related effect: After application at week 0 the treated communities rapidly change treatment dependent. However to the end of the experiment the treated and the control get similar again, which indicates a 'recovery'.

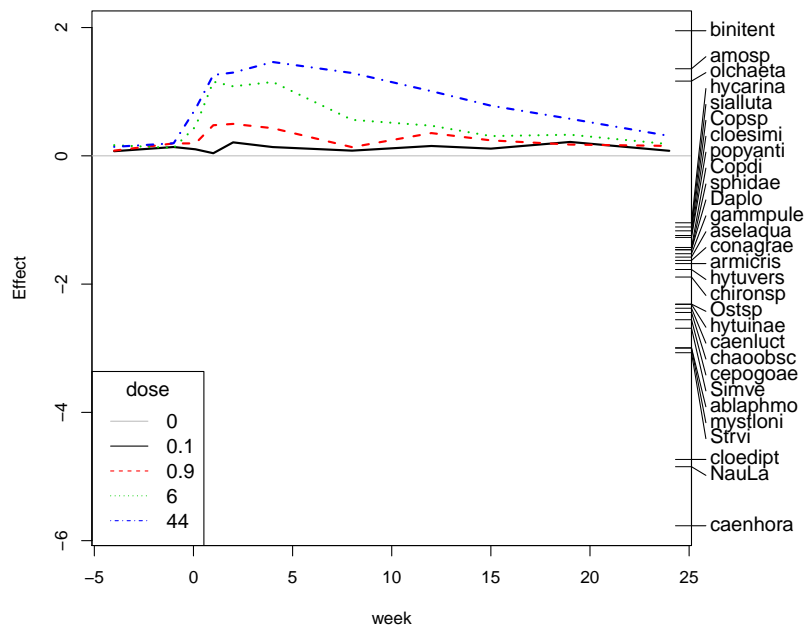


Figure 7.1: Principal response curves (PRC) with species weights for the pyrifos data set indicating effects of the insecticide on the invertebrate community.

On the right-hand side we see the species names and their scores. The more extreme the scores the more this species contributed to the plotted pattern. However, you cannot directly infer from these species scores which species are more susceptible. For example *Gammarus pulex* (gammupule) has a relatively low scores, although it's response pattern (Figure 7.2) shows a strong response, but with no recovery. PRC displays global pattern in the community, but the pattern of *G. pulex* is different from most other species, therefore it gets a lower species score.

We can also look at the numerical output⁴ for this plot using the summary method:

```
pyrifos_prc_sum
```

```
##
## Call:
## prc(response = pyrifos, treatment = dose, time = week)
## Species scores:
##      Simve      Ostsp      NauLa      Strvi binitent caenhora
##      -2.688     -2.312     -4.847     -3.070      1.951     -5.768
## caenluct cloedipt hytuihae ablaphmo cepogoe chaoobsc
##      -2.376     -4.734     -2.316     -2.993     -2.555     -2.442
```

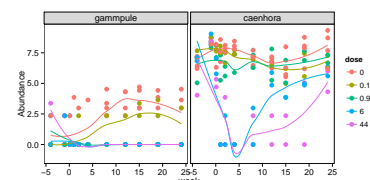


Figure 7.2: Responses of *G. pulex* and *C. horaria* to chlorpyrifos.

⁴ Only a shortened output is given here.

```
## mystloni
## -2.998
##
## Coefficients for dose + week:dose interaction
## which are contrasts to dose 0
## rows are dose, columns are week
##      -4      -1      0.1      1      2      4
## 0.1 0.07218 0.1375 0.1020 0.04068 0.2101 0.1364
## 0.9 0.08106 0.1935 0.1936 0.47699 0.4977 0.4306
## 6   0.16616 0.1232 0.4539 1.15638 1.0835 1.1511
## 44  0.13979 0.1958 0.7308 1.26088 1.2978 1.4627
```

The output of `prc()` gives us more detailed information about the RDA model:

```
pyrifos_prc
```

```
## Call: prc(response = pyrifos, treatment = dose,
## time = week)
##
##              Inertia Proportion Rank
## Total          288.992      1.000
## Conditional     63.349      0.219   10
## Constrained     96.684      0.335   44
## Unconstrained  128.959      0.446   77
## Inertia is variance
##
## Eigenvalues for constrained axes:
##  RDA1  RDA2  RDA3  RDA4  RDA5  RDA6  RDA7  RDA8
## 25.28  8.30  6.04  4.77  4.15  3.86  3.59  3.33
##
## Eigenvalues for unconstrained axes:
##  PC1  PC2  PC3  PC4  PC5  PC6  PC7  PC8
## 17.16  9.19  7.58  6.06  5.73  4.84  4.52  4.10
## (Showed only 8 of all 77 unconstrained eigenvalues)
```

We see that 21.9 % of the variance can be attributed to time (Conditional), 33.5 % can be explained by the treatment regime (Constrained) and 44.6 % of residual variance (Unconstrained), which cannot be explained by time and treatment.

The first RDA axis has an eigenvalue of 25.3. If we divide this eigenvalue by the sum of all eigenvalues, we get the proportion of explained variance which is displayed on the first axis⁵:

⁵ `rda()` (and therefore also `prc()`) returns a huge object with all kind of information stored in it. See `?cca.object` for the internal structure. Here I directly access the eigenvalues from this object

```
pyrifos_prc$CCA$eig[1]/sum(pyrifos_prc$CCA$eig) * 100
## RDA1
## 26.15
```

The significance of the PRC diagram can be tested via permutations. However observations from a experimental ditch are not independent, since the same ditch was measured repeatedly during the experiment. We have to take this into account: each ditch represents a time-series. We will permute the whole series of one ditch, keeping the temporal order (see Tab. 7.1).

To setup such a permutation scheme we use the `permut` package, which is automatically loaded with `vegan`:

```
control = how(plots = Plots(strata = ditch, type = "free"),
              within = Within(type = "none"), nperm = 99)
```

With this setup we can create a permutation matrix. Each row therein is one permutation, the values are the row numbers of the original data set.

```
set.seed(1234)
permutations <- shuffleSet(nrow(pyrifos), control = control)
```

This can be passed to `permutest`, testing the first eigenvalue of our model.⁶

```
mod_perm <- permutest(pyrifos_prc,
                     permutations = permutations,
                     first = TRUE)

mod_perm

##
## Permutation test for rda
##
## Call: prc(response = pyrifos, treatment = dose,
## time = week)
## Permutation test for first constrained eigenvalue
## Pseudo-F: 15.1 (with 1, 77 Degrees of Freedom)
## Significance: 0.01
## Based on 99 permutations under reduced model.
```

We see that our first axis explains a statistically significant proportion of variation (Fig. 7.3). The minimum p-value that we could get is 0.01 (=1/no. of permutations).

Table 7.1: 3 ditches observed for 4 weeks and a possible permutation.

Week	Ditch	Perm
1	1	3
2	1	3
3	1	3
4	1	3
1	2	1
2	2	1
3	2	1
4	2	1
1	3	2
2	3	2
3	3	2
4	3	2

This sets up our permutation scheme:

plots We will permute ditches, without any restrictions.

within But within one ditch there will be no permutations.

nperm We want 99 permutations.

⁶ `vegan` is in active development and at the moment the `permut`-package isn't fully hooked up. Therefore we have to create a permutation matrix beforehand. In the future we will be able to pass the permutation scheme directly into `vegan` functions.

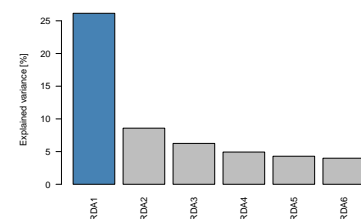


Figure 7.3: Proportion of explained variance of the first 6 RDA-axes. Note that treatment and time were factors and (internally) dummy-coded, therefore we have a total of 44 axes.

Effects per week

After looking at the overall treatment effect, we may want to look at effects at individual time-points. We follow here ⁷ and use the ln-transformed nominal dose as continuous explanatory variable ⁸.

```
dose_c <- log(20 * as.numeric(levels(dose))[dose] + 1)
```

No we can write a for-loop and compute for every week a RDA and a permutation test ⁹.

```
rdas <- NULL
for (i in levels(week)) {
  rdas[[i]]$rda <- rda(pyrifos[week == i, ] ~
                      dose_c[week == i])
  rdas[[i]]$anova <- anova(rdas[[i]]$rda, by = 'terms',
                          step = 199)
}
```

However there is also convenience function in the getx package:

```
rdas <- rda_per_time(pyrifos, dose_c, week)
```

This returns a very big list: one list entry per week and each entry itself contains two lists: rda(RDA-Model) and anova (permutation test)).

From this we have to extract the information we need. We can use `sapply()` to apply a function to every list entry and return results in a vector.

For example to extract the p-values for each week we can use:

```
sapply(rdas, function(x) x$anova[1, 5])

##      -4      -1      0.1      1      2      4
## 0.437186 0.894472 0.005025 0.000000 0.000000 0.000000
##      8      12      15      19      24
## 0.000000 0.000000 0.030151 0.010050 0.185930
```

Have a look at the object structure to write a custom function to extract the information you need:

```
str(rdas[[1]]$anova)
```

Other methods

Other methods to analyse mesocosm experiments include:

⁷ Van den Brink and Ter Braak, 1999

⁸ But before we have to convert dose from a factor to a numeric vector via `as.numeric(levels(x))[x]`

⁹ First we create an empty object (rdas) that will hold our results. Next we run on a subset of data (based on week) a RDA and permutation test. The results of both are stored as a list entry.

multivariate GLMs ¹⁰ In R: mvabund-package.

trait-based indicators ¹¹ Currently no package, but look at rspear-package.

community endpoints ¹² Use vegan for computations.

¹⁰ Warton et al., 2011; and Wang et al., 2012

¹¹ Liess and Beketov, 2011

¹² Sanchez-Bayo and Goka, 2012

7.3 Species Sensitivity Distributions

```
require(fitdistrplus)
# or
require(drc)
```


R Session Info

```
sessionInfo()

## R version 3.0.2 (2013-09-25)
## Platform: x86_64-pc-linux-gnu (64-bit)
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8
##  [2] LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8
##  [4] LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8
##  [6] LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8
##  [8] LC_NAME=C
##  [9] LC_ADDRESS=C
## [10] LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8
## [12] LC_IDENTIFICATION=C
##
## attached base packages:
##  [1] stats4      splines    grid        stats      graphics
##  [6] grDevices  utils      datasets    methods    base
##
## other attached packages:
##  [1] drc_2.3-7          plotrix_3.5-1
##  [3] nlme_3.1-111       magic_1.5-4
##  [5] abind_1.4-0        gtools_3.1.0
##  [7] alr3_2.0.5         car_2.0-19
##  [9] nnet_7.3-7         MASS_7.3-29
## [11] fitdistrplus_1.0-1 survival_2.37-4
## [13] ggplot2_0.9.3.1    reshape2_1.2.2
## [15] qetx_0.0.1         vegan_2.1-35
## [17] lattice_0.20-24    permute_0.7-4
## [19] knitr_1.5
```

```
##  
## loaded via a namespace (and not attached):  
## [1] codetools_0.2-8    colorspace_1.2-4  
## [3] dichromat_2.0-0    digest_0.6.3  
## [5] evaluate_0.5.1     formatR_0.9  
## [7] gtable_0.1.2       highr_0.2.1  
## [9] labeling_0.2        munsell_0.4.2  
## [11] parallel_3.0.2     plyr_1.8  
## [13] proto_0.3-10       RColorBrewer_1.0-5  
## [15] scales_0.2.3       stringr_0.6.2  
## [17] tools_3.0.2
```


Bibliography

Legendre, P. and Legendre, L. (2013). *Numerical ecology*. Elsevier, Amsterdam; Boston.

Liess, M. and Beketov, M. (2011). Traits and stress: keys to identify community effects of low levels of toxicants in test systems. *Ecotoxicology*, 20(6):1328–1340.

Sanchez-Bayo, F. and Goka, K. (2012). Evaluation of suitable endpoints for assessing the impacts of toxicants at the community level. *Ecotoxicology*, 21(3):667–80. pdf RS.

Van den Brink, P. and Ter Braak, C. (1999). Principal response curves: Analysis of time-dependent multivariate responses of biological community to stress. *Environmental Toxicology and Chemistry*, 18(2):138–148.

Wang, Y., Naumann, U., Wright, S. T., and Warton, D. I. (2012). mvabund- an r package for model-based analysis of multivariate abundance data. *Methods in Ecology and Evolution*, 3(3):471–474.

Warton, D. I., Wright, S. T., and Wang, Y. (2011). Distance-based multivariate analyses confound location and dispersion effects. *Methods in Ecology and Evolution*, 3(1):89–101.