

RAD – a FORTRAN program for the study of relative abundance distributions

Version 4.1

Werner Ulrich

Nicolaus Copernicus University in Toruń

Department of Animal Ecology

Gagarina 9, 87-100 Toruń; Poland

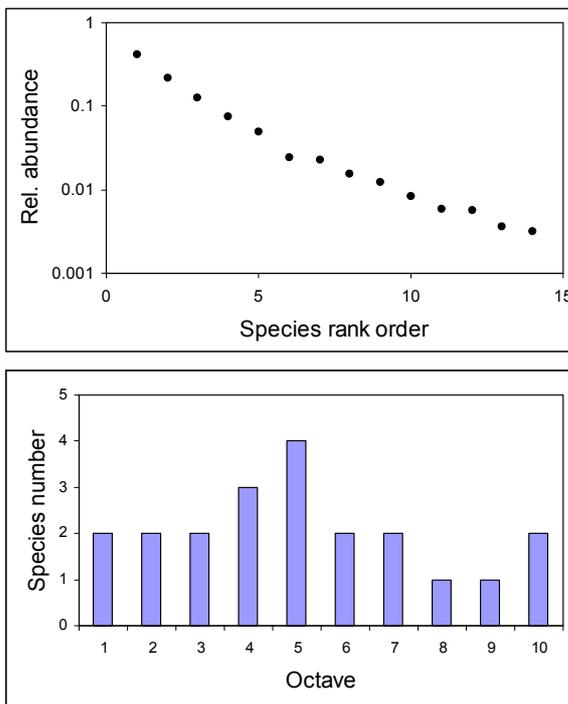
e-mail: ulrichw @ uni.torun.pl

Latest update: 16.11.2003

1. Introduction

All ecological communities are characterized by a set of basic distributions that depend either on species body weights or on species abundances. Among them are the so-called relative

Figure 1: Two ways to visualize the distribution of abundances in an ecological community. Either normalized abundances (absolute species abundances divided through total community abundance) are plotted against species abundance rank order or species numbers per abundance octave against octave. Octaves are \log_2 -transformed densities.



abundance distributions (RADs, often also termed frequency distributions or dominance—rank order distributions) are of major importance. RADs tell something about the distributions of abundances in a community of animals or plants. There are several ways to visualize RADs. Most often used are plots as in Fig. 1. Detailed information about ways of plotting RADs contain Preston (1962), Tokeshi (1990, 1993) and Ulrich (2002).

Because RADs often seem to follow simple geometric shapes ecologists have long sought to understand what causes certain RAD shapes and how to model them. This led to a large number of different theoretical models, a manifold of (often contradicting) attempts to explain observed patterns and to a more or less confusing discussion about the applicability of some models. Overviews about models, theoretical foundations and applications can be found in Preston (1962), May (1975), Sugihara (1980), Magurran (1988), Tokeshi (1990, 1993, 1996), Hubbell (1997, 2001), and Ulrich (2001a, b, c, 2002)

One of the causes for the poor state of art is surely that it proved to be extremely difficult to fit most of the models (especially the more impor-

tant ones) to given data sets. Older work often contains fits by eye and only a few papers deal explicitly with the problem of data fitting (Wilson 1991, Tokeshi 1993, Bersier and Sugihara, 1997, Ulrich 2001b). However, the problem of fitting especially the newer and more realistic stochastic models remains largely unsolved.

RAD is an attempt to solve the problems around data fitting. The program computes and fits nearly all models proposed so far. Additionally it returns a number of additional statistics for instance diversity and evenness indices, goodness of fit measures, and estimates of variances and confidence limits. The program also allows computing truncated distributions and takes automatically predefined samples out of the computed model distributions.

2. Theoretical foundations

Models of relative abundances can be classified into two main groups: deterministic distribution based models and stochastic models based

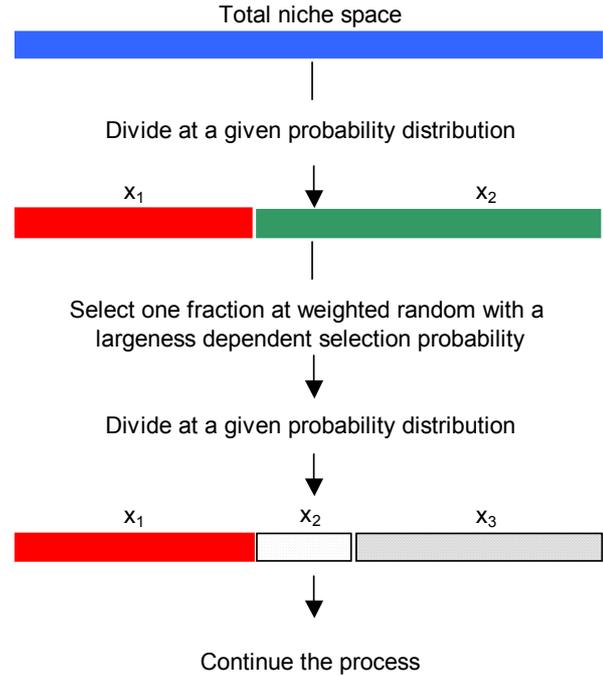


Figure 2: Niche apportionment and sequential break-age models generate relative abundances via a two step process. In a first step total niche space is divided at a given probability distribution. A second step then chooses a fraction from a given selection function and the process continues.

on assumed patterns of resource use. To the first group belong most of the older models like the

Table 1: Models actually included in the program and their computing algorithms. The Fractal model is identical with the stochastic Zipf-Mandelbrot model. The MacArthur fraction is computed via the power fraction model with $z = 1$.

Stochastic Model (program abbreviation)	Reference	Equivalent or similar deterministic model (program abbreviation)	Division probability distribution at	Select niche fraction at
Sugihara fraction (sug)	Sugihara (1980)	Lognormal (log, norm)	$z : 1-z ; 0.5 \leq z < 1$	Random
Power fraction (pow)	Tokeshi (1996)	Lognormal (log, norm)	Random	rx_i^z $-\infty < z < +\infty$
Random fraction (ranf)	Tokeshi (1990)		Random	Random
MacArthur fraction (pow, $z=1$)	Tokeshi (1990)	Broken stick (bro)	Random	$ranx_i$
Random assortment (rapo, rane)	Tokeshi (1990)	Geometric series (geo), log-series (lser)	$ran^z ; 0 < z < 1$	Always the smallest
Dominance preemption (dpre)	Tokeshi (1990)	Geometric series (geo), log-series (lser)	Random	Always the smallest
Dominance decay (ddec)	Tokeshi (1990)		Random	Always the largest
Particulate niche (par)	MacArthur (1957)		Random	None
Overlapping niche (over)	MacArthur (1957)		$ran_1 - ran_2$	None
k-factor model (kfac)	Ulrich (2001a)	Poisson distributions	random or k-times at random	None
Fractal model	Mouillot et al. (2000), Bell (2000), Frontier (1985)	Zipf-Mandelbrot (zm)	Accumulation of whole branching trees at certain probability distributions	
Stochastic Zipf-Mandelbrot (stoz)	Ulrich (2001a)	Zipf-Mandelbrot (zm)	$1 / (ran+X)^z ; 0 < z < \infty$	None
Normal distribution (norm)		Lognormal (log, norm)	Normal random variable	
Hubbell model	Hubbell (2001)	Lognormal (log, norm)	Random sampling from a birth, death, and species turnover process	

lognormal (Preston 1962), the negative binomial (Pielou 1977), the broken stick (MacArthur (1957), the geometric series (Motomura (1932), the log-series (Fisher et al. 1943), or the Zipf-Mandelbrot and fractal models (Frontier 1985, Bell 2000, Moulliot et al. 2000). The second group contains Sugihara's sequential breakage model and its various modifications (Sugihara 1980, Ulrich 2001a), the niche apportionment models introduced by Tokeshi (1990), the neutral model proposed by Hubbell (1997, 2001), or the variability model of Ulrich (2001c). Table 1 gives a condensed overview over the more important models and the computing algorithms. Fig. 2 shows how a two-step process computes stochastic models by selecting a resource fraction and dividing this fraction into further two resource fractions.

Fig. 3 shows relative abundance—species rank order plots of all of the models RAD computes. We see immediately that many of these models are only of theoretical interest because in practice it will be impossible to distinguish them from others with similar shapes.

In general, we can subsume all models into 4 classes. One class of models result in linear shapes similar to the ones of simple exponential functions. To this class belong the geometric series, the random assortment model, the dominance pre-emption (and more or less the dominance decay) and, as a descriptor of samples, the log-series and the negative binomial. The second class contains all models that generate S-shaped distributions, or, if we deal with truncated models, shapes that lack the lower curvature. To this class belong the lognormal, the random, power, and MacArthur fraction and the broken stick. The third class contains models that result in a straight line when using a double log-plot. They contain all models that are based on allometric functions. The last class is made up by models with a lower but without an

upper bend. They are generated by Poisson type functions and contain the k-factor, the overlapping niche and the particulate niche model.

In principle, we may take from each of these classes one appropriate model and seek whether real RADs can be described by this model. Fig. 3 D shows furthermore that the power fraction model proposed by Tokeshi (1996) may even serve as a general descriptor of RADs because it can mimic all distributions of the first two classes by varying only one shape generating parameter. Additionally it seems that classes 3 and 4 are only very rarely found in nature. However, detailed studies to confirm this speculation are still missing.

3. The Program

3.1. Computing relative abundance distributions

3.1.1. Models included

Actually the program computes the following models:

A) Deterministic and semi-deterministic models

a) Broken stick (MacArthur 1957)

The model gives relative abundances D_i from the following series

$$P_k = \frac{1}{S} \sum_{i=0}^{k-1} \frac{1}{S-i}$$

b) Geometric series (Motomura 1932)

The geometric series is computed as

$$D_i = z(1-z)^i$$

The value of z may range between 0 and 1.

c) Lognormal (Preston 1962)

The next three distributions do not allow computing relative abundances directly. They are instead estimated from expected species numbers per octave (see above). Estimation is based on a random assignment (at a log-scale) of abundances inside the abundance range per octave R . For the

lognormal species numbers are computed from

$$S(R) = \exp(-z^2(R - R_{mean})^2)$$

The value of z may range between 0 and 1. The canonical lognormal has a value of $z = 0.2$. R_{mean} may vary between 1 and the maximum number of octaves c ($=\text{lb}(D_1 / D_{min})$). To be sure to get not a truncated distribution R_{mean} should be near $c / 2$.

d) **Negative binomial** (Anscombe 1950)

Species numbers are computed from

$$p_i = \frac{\Gamma(z+i)}{i! \Gamma(z)} \frac{X^i}{(1+X)^{z+i} [1-(1+X)^{-z}]}$$

with $0.00001 < X < \infty$ and $0.00001 < z < \infty$. p_i is the probability that a species has exactly i individuals.

e) **Log-series** (Fisher et al. 1943)

Species numbers are computed from

$$n_i = z \frac{X^i}{i}$$

with $1 \leq X < \infty$ and $0.00001 < z < \infty$. n_i is the frequency that a species has exactly i individuals,

Lognormal, negative binomial and log-

series models are therefore computed from grouped distributions. From these the species numbers per octave are generated. A random process then generate relative abundances inside each octave. These last three models are therefore “semi-stochastic” and a RUNS-value above 1 (see below) returns an abundance variance value for each species.

f) **Zipf-Mandelbrot** (Frontier 1985)

The model gives the relative abundance D for species i from

$$D_i = (i + X)^{-z}$$

with $-0.99999 \leq X < \infty$ and $0 < z < \infty$.

B) **Stochastic models**

a) **Generalized random assortment**

(Tokeshi 1990, Ulrich 2001a):

$$D_i = D_{\min(i-1)} \text{ran}^z$$

ran denotes always a random variable. In the original model is $z = 1$. The value of z may range between $0.00001 \leq z \leq 1$ (*rapo*) or $-0.00001 > z > -10$ (*rane*). An *rapo*-input of $z = 0$ is equivalent with the

Table 2: An example of the batch file that controls model generation in RAD. The number of models computed during one program run is unlimited. Ordering of input values must be the same as in the headline.

Species	Sample	Runs	Density	z	X	Com	Type of S	Samp of S	Diverspec	Model
500	0.000001	10	0	0.1	0	0.7	spec	no	sha	pow
20	10	50	100000	0.75	0.085	1	ind	yes	simp	sug
100	0.001	10	2000	0	0	1	ind	no	evar	ranf
20	0.5	200	0	0.1	0	1	spec	yes	sha	rapo
20	1	50	0	-0.9	0	1	spec	yes	sha	rane
20	1	50	0	0.7	0	1	spec	yes	sha	over
20	1	50	0	0.5	0	1	spec	yes	sha	ddec
20	1	50	0	0.5	0	1	spec	yes	sha	dpre
20	1	50	0	0.5	0	1	spec	yes	sha	par
20	1	50	0	5	0.5	1	spec	yes	sha	kfac
20	1	1	0	2.5	-0.5	1	spec	yes	sha	zm
20	1	50	0	3.7	0.9	1	spec	yes	sha	stoz
20	1	1	0	0	0	1	spec	yes	sha	bro
20	1	50	0	0.2	5	1	spec	yes	sha	log
20	1	10	0	0.1	2	1	spec	yes	sha	norm
20	1	50	0	5	0.99	1	spec	yes	sha	lser
20	1	1	0	0.1	0	1	spec	yes	sha	geo
20	1	1	0	0.5	100	1	spec	yes	sha	nbin
20	1	10	1000	200	0.1	1	spec	yes	sha	hub

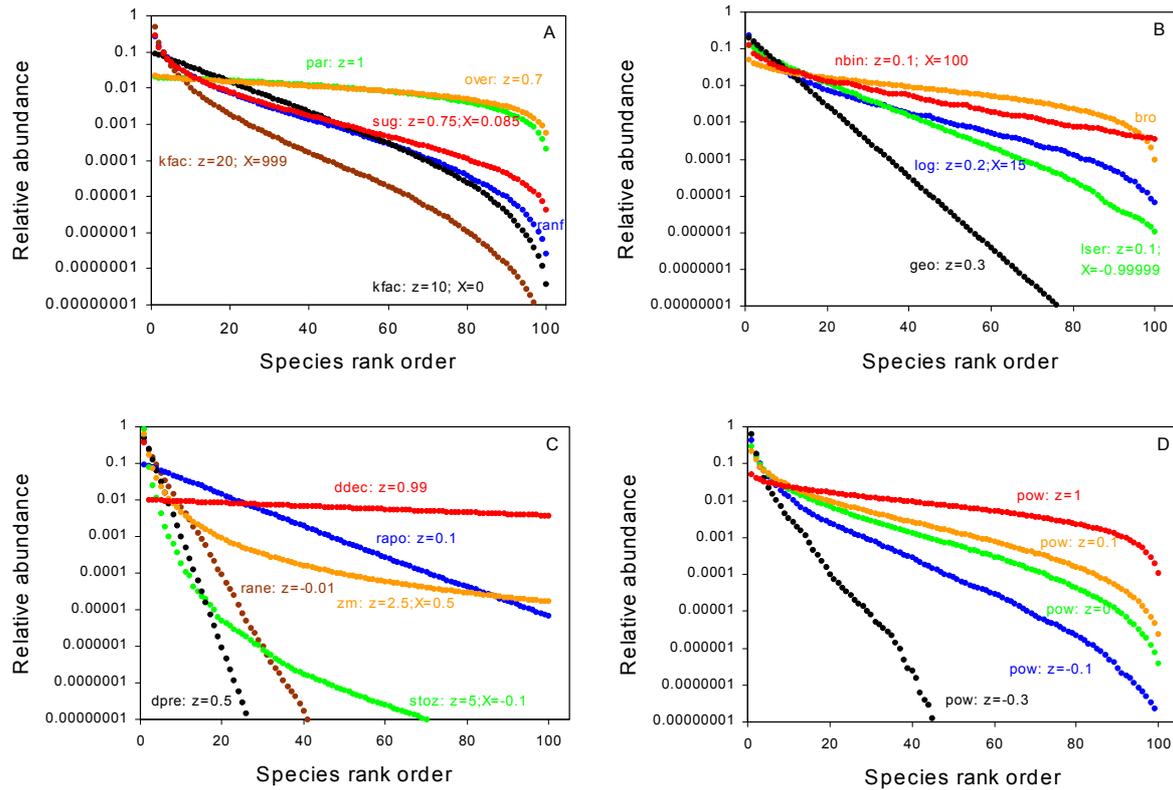


Figure 3: Examples of relative abundance—species rank order plots of models included in RAD. Given are also the respective parameter values for z and X . A: contains most of the more or less S-shaped distributions; B gives the classical deterministic or semi-deterministic models; C contains exponential or allometric models. D shows five examples of the power fraction model of Tokeshi (1996).

- original version of the model.
- b) **Generalized dominance pre-emption** (Tokeshi 1990)
See Tab.1. The value of z may range between $0.00001 \leq z \leq 0.99999$.
- c) **Generalized overlapping niche** (MacArthur 1957)
See Tab. 1. The value of z may range between $0.00001 < z < 10$.
- d) **Generalized particulate niche** (MacArthur 1957)
See Tab. 1. The value of z may range between $-100 < z < 100$.
- e) **MacArthur fraction** (Tokeshi 1990)
See Tab. 1, Thee model is computed via the power fraction model with $z = 1$.
- f) **Generalized Sugihara fraction** (Generalized sequential breakage)

- (Sugihara 1980, Ulrich 2001a)
- The program computes the model either via a fixed breakage value z ($0.5 \leq z < 1$) ($X=0$) or from a normally distributed breakage value with a mean of z (Sugihara 1980, Tokeshi 1996). In the latter case X is the input for the distribution variance. A triangular breakage probability is approximated by a variance input of $X = 0.085$. A simple linear random distribution of z -values is generated after a z and X input of 0. $z = 0.66$ and $X = 0$ is nearly identical with a canonical lognormal. The original Sugihara fraction has $z = 0.75$ and $X = 0.085$.
- g) **Random fraction** (Tokeshi 1990)
See Tab. 1. No input parameter is necessary.

- h) **Generalized dominance decay** (Tokeshi $D_i = ran^{z(X)}$ 1990). See Tab. 1. The value of z may range between $0.00001 \leq z \leq 0.99999$.
- i) **K-factor model** (Ulrich 2001a)
For a simple Poisson distribution the input has to be $z = 0$ and $X = 0$. For z larger than 0 ($1 \leq z \leq 100$) it is possible to work with a fixed number of k-factors $z(X)$ for each species ($X = 999$), with a linear random number $z(X)$ of k-factors ($X = 0$), or with a normal distributed random number $z(X)$ of k-factors (typical range of variances between 0.05 and 1).
- j) **Power fraction** (Tokeshi 1996)
See Tab. 1. The single parameter z can take all values between $-\infty$ and $+\infty$. For $z = 0$ the model is equivalent to a random fraction, $z = 1$ results in a MacArthur fraction.
- k) **Stochastic Zipf-Mandelbrot** (Mouillot et al. 2000, Bell 2000, Ulrich 2001a)
See Tab. 1. Both model parameters z and X have to be larger than 0.
- l) **Stochastic normal and lognormal**
This is an alternative to *log*. Either a simple normal distribution is computed (from a normal distributed random variable *ran*; input option: $X = 0$) or an exponential normal (from $\ln(ran)$, input option: $X = 1$), or a lognormal (from $\exp(ran)$; input option: $X = 2$).
In the latter case the single shape generating parameter z (the variance) is roughly $1/(2a)$ (a being the lognormal parameter)
- m) **Hubbell model (Hubbell 1997, 2001)**
This model stems from Hubbell's introduction of a class of neutral macroecological models that predict local community structure of ecologically similar species

from local birth and death processes as well as emigration to and immigration from a regional metacommunity. Local communities are assumed to be always saturated with individuals (zero-sum game) and the individuals of all species are treated identically. The model has two main input parameters. The first is the local abundance (the sample size) *Density* (> 1). The next is Hubbell's fundamental biodiversity number z (> 0.1) that is approximately identical with the metacommunity alpha diversity). Additionally, a third parameter X ($0 \leq z \leq 1$) can be defined that triggers dispersion limitation (Hubbell 2001). For $X = 0$ no dispersion limitation occurs. The Hubbell module uses the species generator approximation given by Hubbell (2001). Because *hub* uses individuals instead of species to generate communities its philosophy differs from all previous models although the shapes that are generated are very similar to the one of the k-factor model computed from log-Poisson distributions. As a default the maximum species number is set to 2000.

All models can either be computed as complete or as truncated distributions. In the later case the least abundant species will be missing. The latter case resembles various forms of the class of composite models proposed by Tokeshi (1990). Computing algorithms may be inferred from Tab. 1 and Fig. 1.

3.1.2. Parameters required

RAD allows either giving all parameters required manually or by reading a batch file of which an example with typical parameter settings is printed out by the program (*Rad.txt*; other file-

names are accepted). Tab. 2 shows this file. The file must be a text file but no formatting or exact spacing is required. Only the ordering inside a line has to be kept. The number of input lines is unlimited, therefore a large number of assemblages might be computed at one program run..

1. *Model*: Input is the model abbreviation given in Tabs. 1 and 2.
2. *Species*: Species numbers are only limited by computer memory and CPU time. For species numbers above 1000 model computing may take several hours. As a default the maximum assemblage size to be generated is 2000 species.
3. *Runs*: Give the number of replicates for each model. Stochastic models result for

each computational run in a different output. RAD computes for each species a range of relative densities and the variance in abundance. Deterministic models have a variance of 0 and it makes no sense to give runs values above 1.

4. *Density*: With this option a maximum density difference D (the quotient of most and least abundant species) can be chosen. This option is an alternative for the computation of truncated distributions since all species with relative abundances less than $1 / D$ are left out. The input of 0 sets the maximum density difference to 10^{16} , a value equivalent to a maximum of 53 binary density classes (octaves).

Table 3: Assemblage.txt for a truncated community of a total of 30 species following a canonical log-normal.

Species	Density	SampSize	Runs	Diver	Model	Const z	Const X	TruncFa	Sampl	FacSpec	
30	1E+16	1	20	sha	log	0.2	10	0.7	spec	yes	
	Species	max.DD	No.Oct.	Div	DivSD	Even	EvenSD	Alpha	SD	SD-SD	
Assemblage	21	303	9	2.05	0.084	0.673	0.027	3.93	2.328	0.073	
Sample	21	303	9	2.053	0.084	0.674	0.028	3.93	2.328	0.073	
	rel. Dens. S1	rel. Dens. S	50%Slop	SlStDev	50%Int	IntStDe	15%Spe	15%Pro	15%SpZM	15%PrZMM	95%CL
Assemblage	0.36823	0.001215	-0.244	0.01	10.618	0.048	4	0.63281	0	0	0.08684
Sample	0.369186	0.001218	-0.244	0.01	10.618	0.048	4	0.63281	0	0	0.08684
Community			Sample								
* Mean	STDev	Sp./Class	Mean	STDev	Sp./Class						
0.3682299	0.044157	3	0.369186	0.044258	3						
0.17414249	0.027661	3	0.174594	0.027726	3						
0.12246525	0.020008	3	0.122785	0.020069	3						
0.08717472	0.01538	3	0.087402	0.01542	3						
0.05353394	0.010299	3	0.053673	0.010321	3						
0.04376933	0.010549	2	0.043883	0.010574	2						
0.03020834	0.005526	2	0.030287	0.00554	2						
0.02461715	0.004542	1	0.024682	0.004556	1						
0.02069569	0.002625	1	0.02075	0.002633	1						
0.01653891	0.002389	0	0.016582	0.002397	0						
0.01338103	0.003161	0	0.013416	0.003171	0						
0.01010528	0.001671	0	0.010132	0.001677	0						
0.00778605	0.001062	0	0.007806	0.001065	0						
0.00600555	0.001156	0	0.006021	0.001158	0						
0.0053364	0.001074	0	0.00535	0.001076	0						
0.00352168	0.00071	0	0.003531	0.000712	0						
0.00293191	0.000613	0	0.00294	0.000615	0						
0.0025599	0.000467	0	0.002567	0.000469	0						
0.00181867	0.000364	0	0.001823	0.000365	0						
0.00136798	0.000243	0	0.001372	0.000244	0						
0.00121472	0.000138	0	0.001218	0.000139	0						

□

5. z : This is the main shape generating parameter of all models.
6. X : This is the second shape generating parameter of some models.
7. Com : The truncation factor that results in shapes similar to the composite model class of Tokeshi. The input $0 \leq Com \leq 1$ gives the fraction of species included in the output. For $Com = 0$ a complete assemblage will be generated.
8. *Type of S*: RAD automatically takes a sample from the computed assemblage. You can either sample species (input: spec) or individuals (input: ind).
9. *Sample*: *Sample* defines the sample size, either as the multiple of species number or as the fraction of the total individual number. The latter is given by *Density*.
10. *Samp of S*: In the case of 'yes' RAD samples at each replicate run exactly the defined number of species as defined in *Sample*. For *Type of S* = 'yes' RAD samples *Sample* times the number of species until this species number is reached. In this case *Sample* may take every value above 0. Note that such a sampling takes place in each of the replicates. For stochastic models larger replicate numbers result in larger sample sizes as defined because the output is a mean values of all species sampled. To get exactly the defined sample size of *Sample* x *Species* you have to run RAD several times with *Runs* = 1. For *Type of S* = 'no' and *Samp of S* = 'yes' the sample size will be *Sample* x *Species* irrespective of the number of species found. For *Type of S* = 'no' and *Samp of S* = 'no' the sample size will be *Sample* x *Density* again irrespective of the number of species.
11. RAD computes measures of diversity H and evenness E of the assemblage generated. You may choose three common diversity indices with it's associated evenness measures (Magurran 1988):

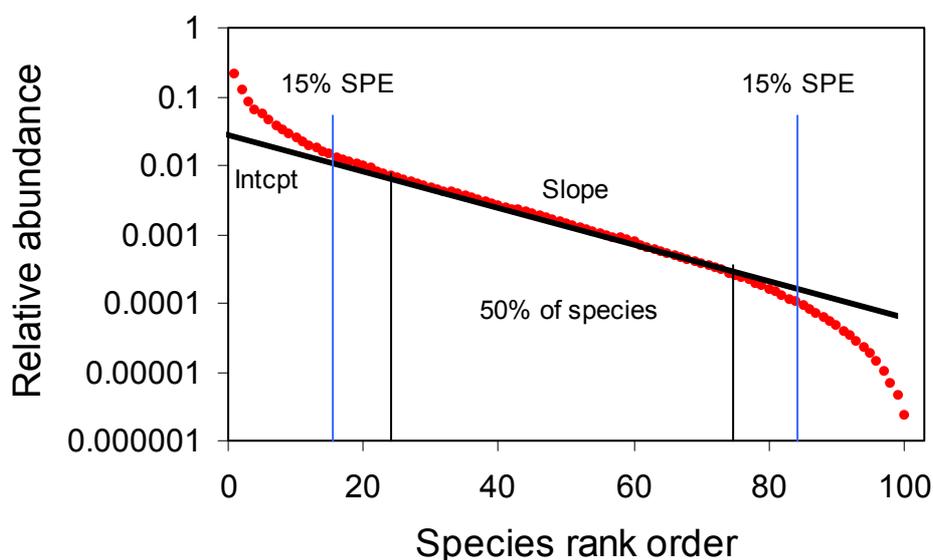


Figure 4: A power fraction model generates a S-shaped frequency distribution. We can compute a regression line through the middle ranking 50% of species. They have the output values 50%Slop and 50%Int together with their standard deviations (SISDev and IntStDe). 15%Spe and 15%SpZM count the number of species of the upper and lower 15% percentile that are above and below (15%Spe) or only above (15%SpZM) the regression line. 15%Pro and 15%PrZM give the probability that this will be the case just by chance. Note that *slope* is a diversity statistics similar to the alpha-index.

a) Smith – Wilson (evar) (Smith and Wilson 1996):

$$E_{var} = 1 - \frac{2}{\pi} \arctan\left(\frac{\sum_{i=1}^S (\ln(p_i) - \frac{\sum_{i=1}^S \ln(p_i)}{S})^2}{S}\right);$$

$$H_{var} = E_{var} \times S$$

b) Shannon – Wiener (sha):

$$H_{sha} = -\sum p_i \ln(p_i); E_{sha} = H_{sha} / \ln(S)$$

c) Simpson (simp):

$$H_{simp} = 1 / \sum p_i^2; E_{simp} = H_{simp} / S$$

E_{var} is equivalent to the arctan transformed Gaussian width. For these three indices RAD gives also standard deviations obtained from the respective values of each replicate.

Additionally RAD computes the α -diversity index (Magurran 1988, Rosenzweig 1995). Note that after a computation with many replicates α of the log-series or the Hubbell model is not identical with z .

do: computing or fitting a given data set or printing examples of two batch files.

Next the program asks about the ending of the output files. RAD prints one standard file *Assemblage???.txt*. With the ending option you define the ending ??? of this file. A carriage return sets ??? to blank.

Then you are asked whether you want to run RAD by hand (option 'no') or via a batch file ('yes'). After the input of 'no' RAD asks stepwise for all the parameters required. At the beginning you should run several times by hand because the program shows all the input parameters with their allowed parameter ranges. With the next option (printing assemblages) you can tell the program only to print the assemblage statistics without printing relative densities. This is a convenient option in the case of large batchfiles were only certain statistics are required.

If you want to run RAD via a batch file you have first to print out examples of these batch files.

3.1.3. Starting the program

After program start RAD asks what to

3.1.4. The output file *Assemblage.txt*

RAD generates a single unformatted

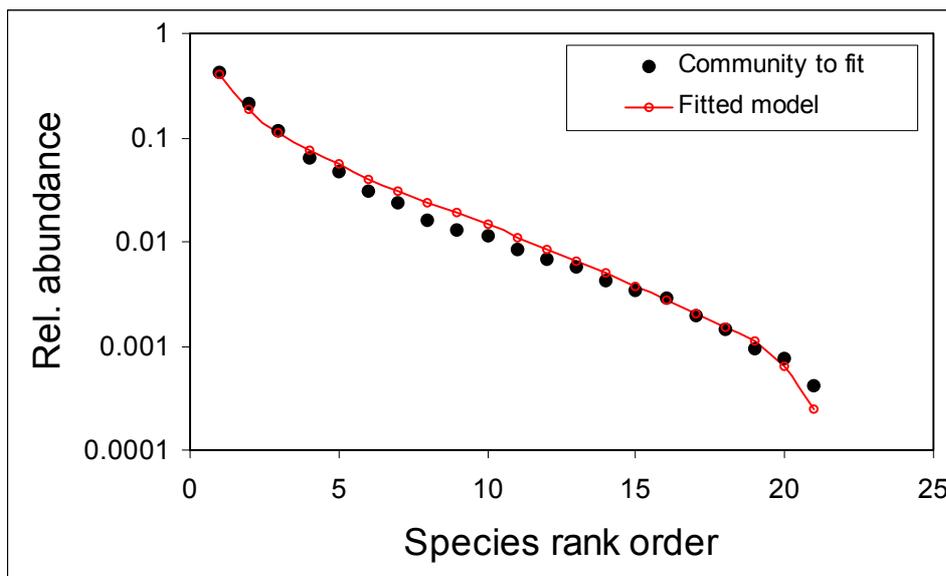


Fig. 5: Example of a community of 21 species following a truncated lognormal distribution (in total 30 species) and a fit by the power fraction model.

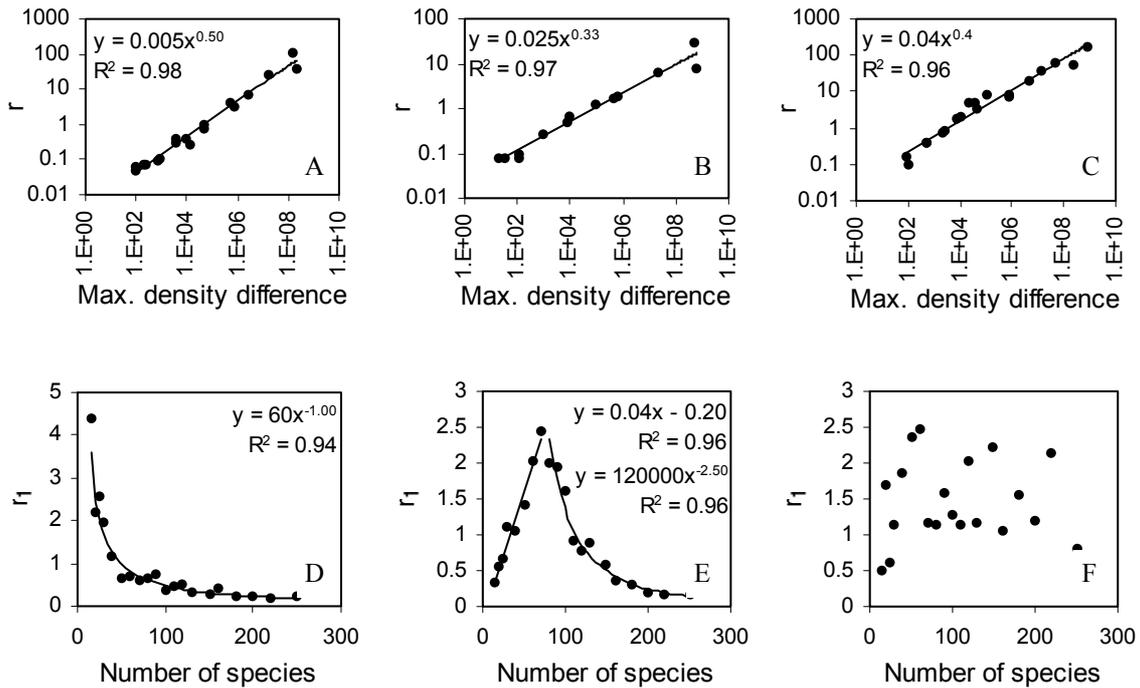


Figure 6. Dependence of the least square test statistics r (termed r_1 after correction for dependence of the maximum density difference) on the maximum density difference (A to C) in the data set and the number of species (D to F). Results of 20 power fraction (A: $z = -2$ to 2 , 50 species each; D: $z = 0.1$), random assortment (B: $z = 0.05$ to 0.8 , 50 species each; D: $z = 0.2$), and stochastic Zipf-Mandelbrot data sets (C: $z = 1$ to 8 , $X = 0$, 50 species each; F: $z = 2$, $X = 0$). The assemblages were computed (means of 100 replicates) and then fitted by the same model. The data points are mean values of 100 fitting procedures each. R^2 : variance explanation of the regression given (Modified from Ulrich 2001a).

output file called *Assemblage.txt*. An example of this file contains Tab. 3

The first two output lines give the initial parameter settings. The next two lines contain basic statistics of assemblage and sample: Number of species, maximum density difference between most (S1) and least (S) abundant species, the number of binary density classes (octaves), diversity (Div) and evenness (Even) indices according to the initial setting (the values in Table 3 refer to the Shannon index) together with their standard deviations (in the case of multiple Runs), the α -diversity statistic, and, at least, the Gaussian weight (together with its standard deviation). The Gaussian weight is the standard deviation of log-transformed relative abundances.

The next two lines contain further descriptive statistics. The first two positions are the relative densities of the most and the least abundant

species. The next four positions give slopes and intercepts (together with their standard deviations) of exponential regression lines through the middle ranking 50% of species (Fig. 4). At last, species numbers (of the upper and lower 15% percentile) of species ranging above and below this regression line are counted (Fig. 4). RAD gives also the probability that these species numbers deviate from the null hypothesis of a random pattern around the regression line. The program estimates therefore deviations from linearity.

The last value M 95%CL computes a mean 95% confidence limit of relative densities for the middle ranking 70% of species. The value of 0.08684 indicates therefore that the mean confidence limit of relative density for these species is about 9% of relative density. Note that standard deviations of relative density are much higher for low abundant species.

Then, the program prints mean densities and standard deviations of all of the species (assemblage and sample). Additionally, the number of species / binary weight class is given, starting from octave 1 ($2^0 = 1$ individuals). This allows immediately drawing a species number—octave plot.

4. Fitting relative abundance distributions

4.1. The fitting algorithms

The natural way to fit a theoretical distribution to a given data set is by least squares. In the case of deterministic distributions this is the most often taken approach (Wilson 1991). In the case of stochastic models the method has not been applied due to the high variance in densities and their density dependence. Instead, Tokeshi's method (Tokeshi 1993, Bersier and Sugihara 1997) using the 95% confidence limit of the model assemblages was preferred. However, this method proved to have only a low discrimination power (see below). It is therefore desirable to look for an

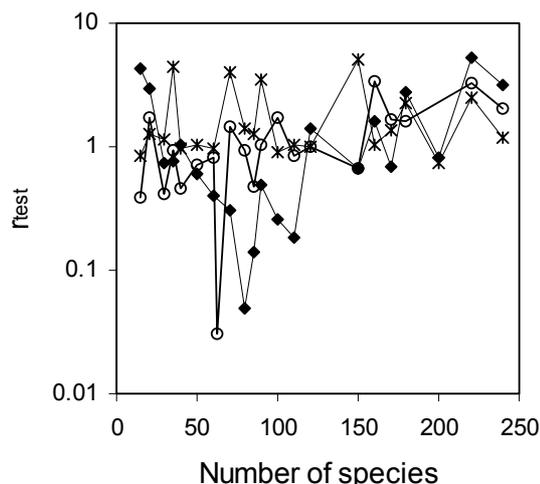


Figure 7. Performance of the test statistic r_{test} after corrections for density and species number differences in dependence on the number of species in the data set to be fitted. 20 data sets were each fitted by the same model and the data points are mean r_{test} values (of 100 replicates each) for each fit. Circles: random assortment, squares: Sugihara fraction, stars: stochastic Zipf-Mandelbrot model.

alternative. Additionally, the method does not allow estimating parameter values. Therefore, in a first step a least square statistic was developed with a better discrimination power.

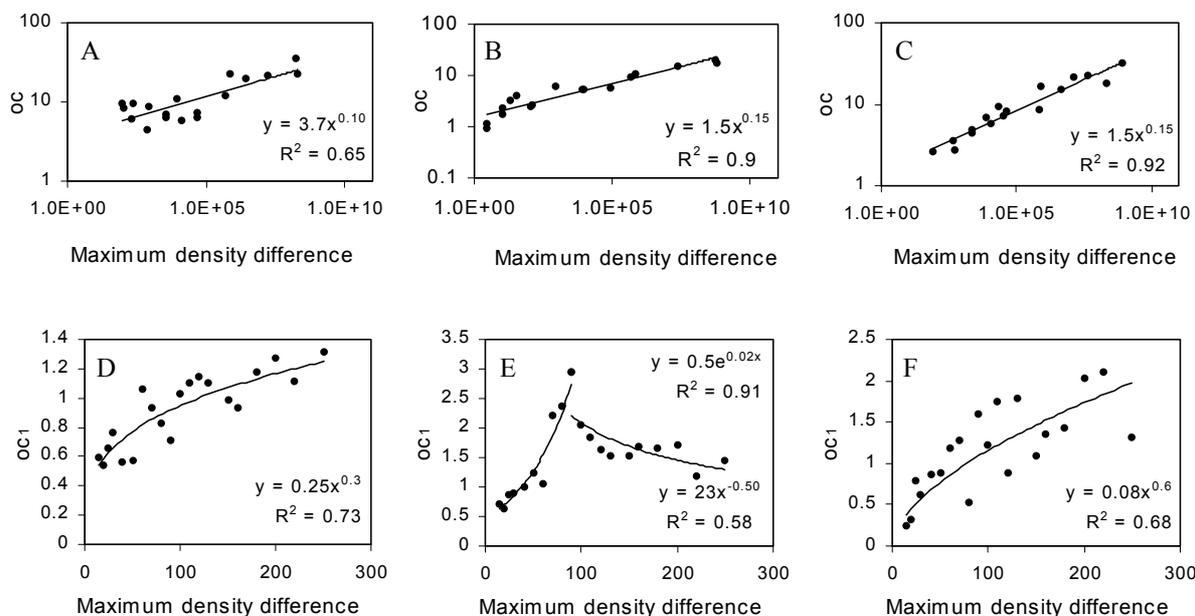


Figure 8. Dependence of the least square test statistics oc (termed oc_1 after correction for dependence on the maximum density difference) on the maximum density difference (A to C) in the data set and the number of species (D to F). Assemblages as in Fig. 6 A, D: power fraction model, B, E: random assortment model, C, F: stochastic Zipf-Mandelbrot model.

Essentially, the fit relies on a least square fits where the Euclidean distances between predicted and observed data points are minimized. The main fitting variable r can then be defined as:

$$r = \sum_{i=1}^S (d_i)^2$$

with d_i being the Euclidean distance between theoretical and empirical ln-transformed densities and S the number of species. In the case of the stochastic models r is computed using mean densities of n replicates of the model. The fitting process runs with different values of the shape generating parameters z and X (by stepwise enclosure) until r reaches a minimum.

This test value is sensitive to the maximum density difference D of the species - defined as the quotient of most and least abundant species - of the data set (Fig. 6) (here data set always refers to an assemblages to be fitted). In the stochastic models low density species effect the value of r overproportional, mainly due to the higher variance. This makes it difficult to compare fits by different models.

Additionally, due to the summation process the test statistic r will depend on the total number of species S when comparing fits from assemblages of different species numbers. At first sight, the quotient r / S should be constant. However, the higher the number of species is the lower is the total variance after a finite number of iterations. Therefore, r will be relatively lower at high species numbers. To use r as a test statistic for comparing the fits of stochastic models at least a correction factor for density is necessary. If one wants to compare r at different species numbers a second correction factor has to be added.

Figs. 6 A to C show the dependence of r on the max. density difference in the data set and Figs. 3 D to F give the dependence of the density corrected r (denoted as r_l) on species number for

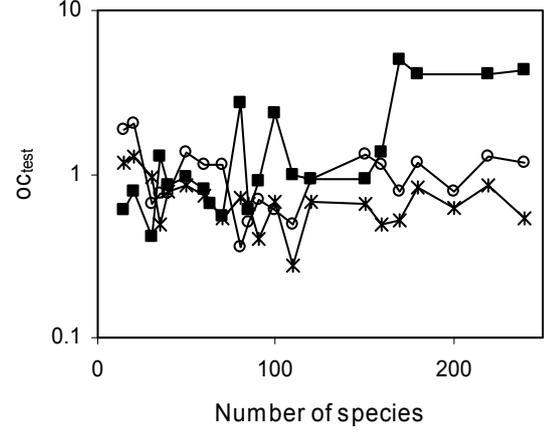


Figure 9. Performance of the test statistic oc_{test} after corrections for density and species number differences in dependence on the number of species in the data set to be fitted. 20 data sets were each fitted by the same model and the data points are mean r_{test} values (of 100 replicates each) for each fit. Circles: random assortment, squares: Sugihara fraction, stars: stochastic Zipf-Mandelbrot model.

the three main types of model shapes: the linear, the allometric, and the S-shaped type. The fourth type described above (generated by the k-factor model and a few others) has very similar fitting properties than the S-shaped type and was subsumed under this.

Introducing the equations given in Fig. 6 into the equation for r (as correction factors) and rearranging conveniently results in the following least square test statistic for stochastic relative abundance distributions:

1. Random assortment type distributions:

$$r_{test} = \frac{r}{(0.0004S + 0.2)D^{0.4}}; \quad (5 \leq S \leq 70)$$

$$r_{test} = \frac{r}{1200 * S^{-2.5} D^{0.4}}; \quad (S > 70)$$

2. Power fraction type distributions

$$r_{test} = \frac{r}{3 * S^{-1} D^{0.5}}$$

3. Zipf-Mandelbrot type distributions

$$r_{test} = \frac{r}{0.03 * D^{0.4}}$$

Of course, the value and the variance of

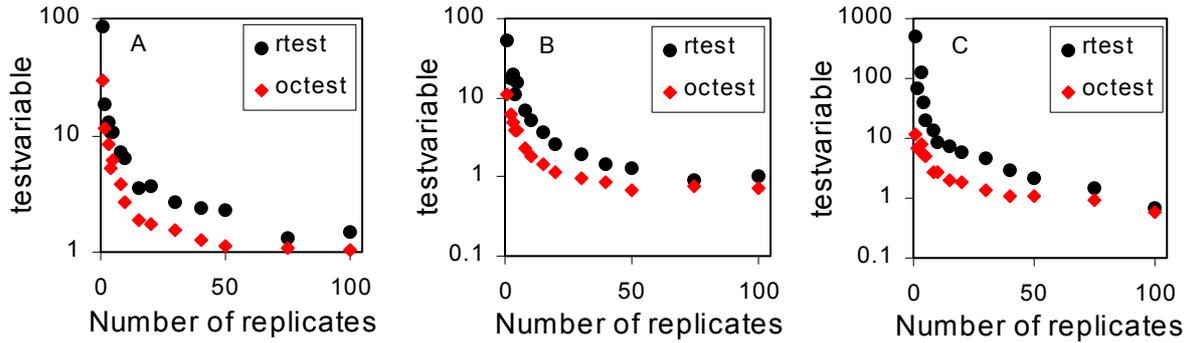


Figure 10: r_{test} and oc_{test} depend on the number of replicates of the stochastic model used for fitting. But it appeared that above 50 replicates the results become more or less stable. A: power fraction model, B: random assortment, C: stochastic Zipf-Mandelbrot. A data set (the same model after 100 replications) was fitted using mean densities after 1 to 100 replications.

r_{test} depends highly on the number of replicates in the data sets but it appeared that the variance dependence of r_{test} is the same for all three types of distributions and that above 50 replicates r_{test} becomes constant (Fig. 10). Good fits are then in every case characterized by values of r_{test} near or below 10 (Figs. 7, 10). Values above 100 can hardly be called fits.

Fig. 7 shows a performance test of r_{test} for all three types of distributions. The test was done by computing 100 replicates each of a Sugi-

hara fraction, a random assortment and a Zipf-Mandelbrot distribution and afterwards fitting these data sets by the same models. For species numbers between 20 and 250 and accompanying density differences between 10 and 10^8 r_{test} was roughly constant ranging between 0.03 and 3.3 (power fraction), 0.14 and 4.3 (random assortment), and 0.73 and 4.4 (stochastic Zipf-Mandelbrot). All data points are inside the range of 2 standard deviations of each model (data not shown).

Table 5: Example of a batchfile for data set fitting. In this case communities of 1000 species are fitted to a data set. zmin and zmax give the program’s default ranges for each model.

Species	Runs	Iteration	IterAss	Density	zmin	zmax	X	Conf Lim	Deviat.F	Fit with	CorrectF	Model
1000	20	10	10	0	-0.1	3	0	0.3	0	a	dens	pow
1000	20	20	10	0	0.51	0.99999	0.085	0.5	0	s	spec	sug
1000	20	10	1	0	0	0	0	0.75	0	a	both	ranf
1000	20	10	1	0	0.01	1	0	0.9	0	a	both	rapo
1000	20	10	1	0	-10	-0.01	0	0.95	0	a	both	rane
1000	20	10	1	0	0.00001	10	0	0.99	0	a	both	over
1000	20	10	1	0	0.00001	0.99999	0	0.95	0	a	both	ddec
1000	20	10	1	0	0.00001	0.99999	0	0.95	0	a	both	dpre
1000	20	10	1	0	-10	10	0	0.95	0	a	both	par
1000	20	10	1	0	0	100	0.1	0.95	0	a	both	kfac
1000	1	10	1	0	0.1	9.99999	0	0.95	0	a	both	zm
1000	20	10	1	0	0.1	10	0	0.95	0	a	both	stoz
1000	1	1	1	0	0	0	0	0.95	0	a	both	bro
1000	20	10	1	0	0.01	0.9	25	0.95	0	a	both	log
1000	20	10	1	0	0.1	10	2	0.95	0	a	both	norm
1000	20	10	1	0	0.01	500	0.9999	0.95	0	a	both	lser
1000	1	10	1	0	0.01	0.5	0	0.95	0	a	both	geo
1000	20	10	1	0	0.01	100	100	0.95	0	a	both	nbin
1000	20	10	1	1000	1	1000	0.2	0.95	0	a	both	hub

A second main fitting statistic uses the number of species per binary density class (octave). As in the case of r_{test} oc_{test} computes squared differences, but in this case those of species numbers per octave of model and data set.

Again this measure depends on the maximum density difference and the total species number (Fig. 8). The respective correction factors are

1. Random assortment type

$$oc_{\text{test}} = \frac{r}{0.075 * e^{0.02S} D^{0.15}}; \quad (5 \leq S \leq 70)$$

$$oc_{\text{test}} = \frac{r}{35 * S^{-0.5} D^{0.15}}; \quad (S > 70)$$

2. Power fraction type

$$oc_{\text{test}} = \frac{r}{0.9 * S^{0.3} D^{0.1}}$$

3. Zipf-Mandelbrot type

$$oc_{\text{test}} = \frac{r}{0.12 * S^{0.6} D^{0.15}}$$

Fig. 9 gives the performance of oc_{test} at various assemblage sizes. Again it appears that good fits have test values below 10. In the case of the S-shaped models oc_{test} still depends slightly on the species number.

RAD computes other statistics to assess the goodness of fit. These are described below in Chapter 2.1.3.

4.2. The input parameters

Computing and fitting procedures are in essential the same. You can run RAD either manually or using a batch file. Many data sets can be fitted during the same program run.

Table 5 shows a typical batch file with all necessary input parameters and the default settings of z , called z_{min} and z_{max} . Note that RAD does not allow an automatic fit for models with two parameters. In those cases several fits with manually modified X-values have to be run.. How-

ever this matters only for the models *nbin*, *zm* and *stoz*. In $\log X$ is the modal octave that is first computed and shown by the program.. *lser* has X-values that are nearly always near 1.

As before the input has to contain the number of species (*Species*). This value will nearly always be identical with the total species number of the data set. If this number is unknown take a sufficiently large value of S. RAD counts the species number of the data set and adjusts the species number. The input *Fit with* = 'a' sets the species number to the number of species found in the data file. In this case *Species* has no implications for the program run until *Density* (the maximum density difference for the fitting model) is set to maximum. If *Species* and *Density* are set too low the fitting process might return unreliable results. Again, the number of replicates (*Runs*) of the model is also necessary.

RAD fits either fully censured communities (data sets) or samples from them. In the first case *Fit with* needs the input 'a'. In the latter case (*Fit with* = s) the total species number of the community has to be known and given in *Species*.

Iteration gives the number of iterations of the model fit. For most models convergence is quite fast and 20 iterations will result in a stable fit. The maximum number of iterations must not exceed the sum of species number and number of replicates of the model.

In *IterAss* the number of replicates (the sample size) of the data set has to be given. This is necessary for computing confidence limits (see below). In most cases *IterAss* will have a value of 1.

Z_{min} and Z_{max} give the range of the main shape generating parameter z . Tab. 5 shows the default settings of the programs but other values inside the above described ranges are allowed (although these will sometimes result in errors

Table 6: Output (file = Assemblage.txt) after the fitting process with the settings of Tab. 5. The data set contained a sample of 20 species.

Col	Species	Density	SampSize	Runs	RepM	Model	Const z	Const X	TruncFa	Modus	Corr-Mod	dens			
1	50	1E+16	20	100	100	rapo	0.3812	0	0	s					
Testvari-able	Testvar Octaves	FR<conf	Rel.Sum R>conf	Chi2-confid	p-value	Kolmog-Smir	p-value	50% Slope	50% Interc	15%Spe	15% Pro	15% SpeZM	15% ProZM		
3.20449	2.25959	0.75	0.049	1.5	0.75	3	0.990	-0.45782	1.03948	0	0	0	0		
Testcom	Fitted Com	0.95 Conf. Limit	Oct. Testcom	Oct. Fitted	Testvar.	Testvar Oct.	Constant z								
0.354792	0.305764	0.05993	2	3	312.9619	81.797	0.01								
0.205842	0.20322	0.039831	1	2	1213.574	4.519	1								
0.114552	0.141243	0.027684	2	1	55.9305	2.712	0.505								
0.083359	0.09908	0.01942	1	3	60.39954	4.971	0.2575								
0.059937	0.071347	0.013984	2	2	524.746	4.519	0.7525								
0.048353	0.052499	0.01029	3	2	176.1115	3.163	0.62875								
0.033452	0.035489	0.006956	3	2	3.067132	1.356	0.38125								
0.027024	0.025357	0.00497	2	2	9.804308	1.356	0.443125								
0.01931	0.018268	0.00358	2	2	16.04448	2.712	0.319375								
0.015078	0.013246	0.002596	1	1	4.459754	1.808	0.412188								
0.010962	0.009732	0.001908	1	0	7.10741	1.808	0.350312								
0.008402	0.00703	0.001378	0	0	6.429427	1.808	0.396719								
0.006319	0.005256	0.00103	0	0	7.273239	1.808	0.365781								
0.004564	0.003788	0.000742	0	0	3.717461	1.356	0.388984								
0.00314	0.002383	0.000467	0	0	3.841184	1.808	0.373516								
0.002066	0.001733	0.00034	0	0	3.204485	2.26	0.38125								
0.001384	0.001233	0.000242	0	0	0	0	0								
0.000736	0.000878	0.000172	0	0	0	0	0								
0.000466	0.000643	0.000126	0	0	0	0	0								
0.000261	0.000467	9.15E-05	0	0	0	0	0								

communications of the program). For a good and faster model fit it will often be better to define more narrow ranges. X will not be changed during program run. In cases where X is important several runs with different X values have to be performed until a sufficiently good fit is reached.

In *ConfLim* the x% confidence limit probability is defined. RAD computes for each species the density range inside this limit and compares the respective data set density with this limit (see below). Allowed input values are (0.3, 0.7, 0.9, 0.95, and 0.99). They refer to confidence limit factors of (0.352, 0.6, 1.17, 1.645, 1.96, and 2.576). For other *ConfLim*-values RAD takes the default probability value of 0.95. Instead of this a fixed confidence limit factors can be given (Input *ConfLim* = 0 and *Deviat.F* = value). Typical values should be between 0.5 and 3.

In the case of deterministic or semideterministic models (*zm*, *log*, *nbin*, *lser*, *bro*, and *geo*) RAD simulates a confidence limit CL by an empirically derived function. For this CL and density of stochastic models were compared. This resulted in the following approximation:

$$CL = factor \frac{0.33D^{0.9}}{\sqrt{n}}$$

Here, D is the density of a species and n the number of replicates of the model.

At last, the correction factor may be defined. This refers to the density and species number dependence of r and oc. With *CorrectF* =dens, only the density dependence is corrected. This is appropriate when comparing different communities with the same species numbers. To compare communities with different species numbers but similar density ranges the input *spec* is indicated. To correct

for species number and density *CorrectF* must be ‘both’.

4.3. The Output

RAD again produces the output file ‘Assemblage.txt’, of which Tab. 6 shows an example.

The first two file lines repeat the input values and give the value of z for which the given model best fits the data set. Col refers to the column of a data matrix. If you have only one column Col will have the value 1.

The next two output lines first contain r_{test} and oc_{test} . Then, the frequency of species ranging inside the $x\%$ confidence limits are given. Tokeshi (1990) introduced this method for fitting relative abundance distributions (see for a modification also Bersier and Sugihara 1997). In principle this method is very simple. It compares the relative abundance f of species i of the data set with the mean density of the i -th species of the fitting model using the $x\%$ confidence limit:

$$Con_i(x\%) = \mu_i \pm \frac{r(x\%)\sigma_i}{\sqrt{n}}$$

Again, n is the number of replicates of the model. If more than say 1 or 5% of species range outside this limit, the model does not fit the data set. We see, however, immediately the problem with this approach. First of all, the method does not take systematic deviations into account. Although all species may range inside the 5% confidence limit, their relative abundances may all be too low or too high. Secondly, the quality of data sets might be very different. In one case 100 specimens had been sampled, in another 100000. We have therefore to include also confidence limits for the data set abundances. RAD does this by modifying the above equation to incorporate the (estimated) sample accuracy of the data set n_{data} .

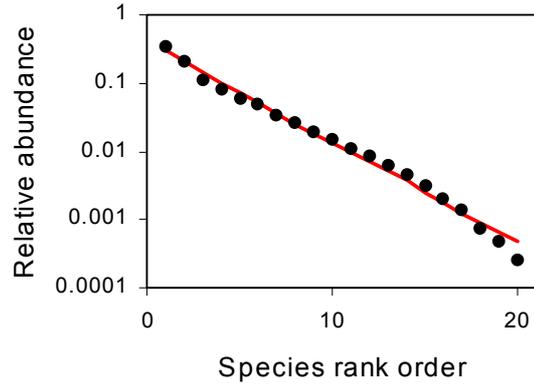


Fig. 10: Data set and model species of the example of Tab. 6.

This is the assumed number of replicates of the data set that has to be estimated from the sample size (see Ulrich 2001b for a detailed discussion of this point).

$$Con_i(x\%) = \mu_i \pm \frac{r(x\%)\sigma_i}{\sqrt{n}\sqrt{n_{\text{data}}}}$$

It then computes the frequency of species ranging inside this confidence limit ($FR < conf$). Additionally, a measure for the magnitude of deviation is given as the sum of deviations of those species ranging outside the $x\%$ confidence limit.

$$R_{Sum,i>conf} = \sum \frac{Con_i(x\%)}{x_{Model}}$$

In a next step RAD computes the probability (via a χ^2 -test) whether the ordering of deviations of those species ranging outside the $x\%$ confidence limit deviates from the null hypothesis of a random distribution around the model mean (χ^2 *confid* and *p-value*). In the example of Tab. 6 the probability that the deviators are distributed non-randomly is greater than 75% but less than 90%.

At least a Kolmogorov-Smirnov test gives the probability that the ordering of relative abundances of all data set species in relation to the model species deviates from the expectation of a random ordering. *KolmogSmir* gives the number of

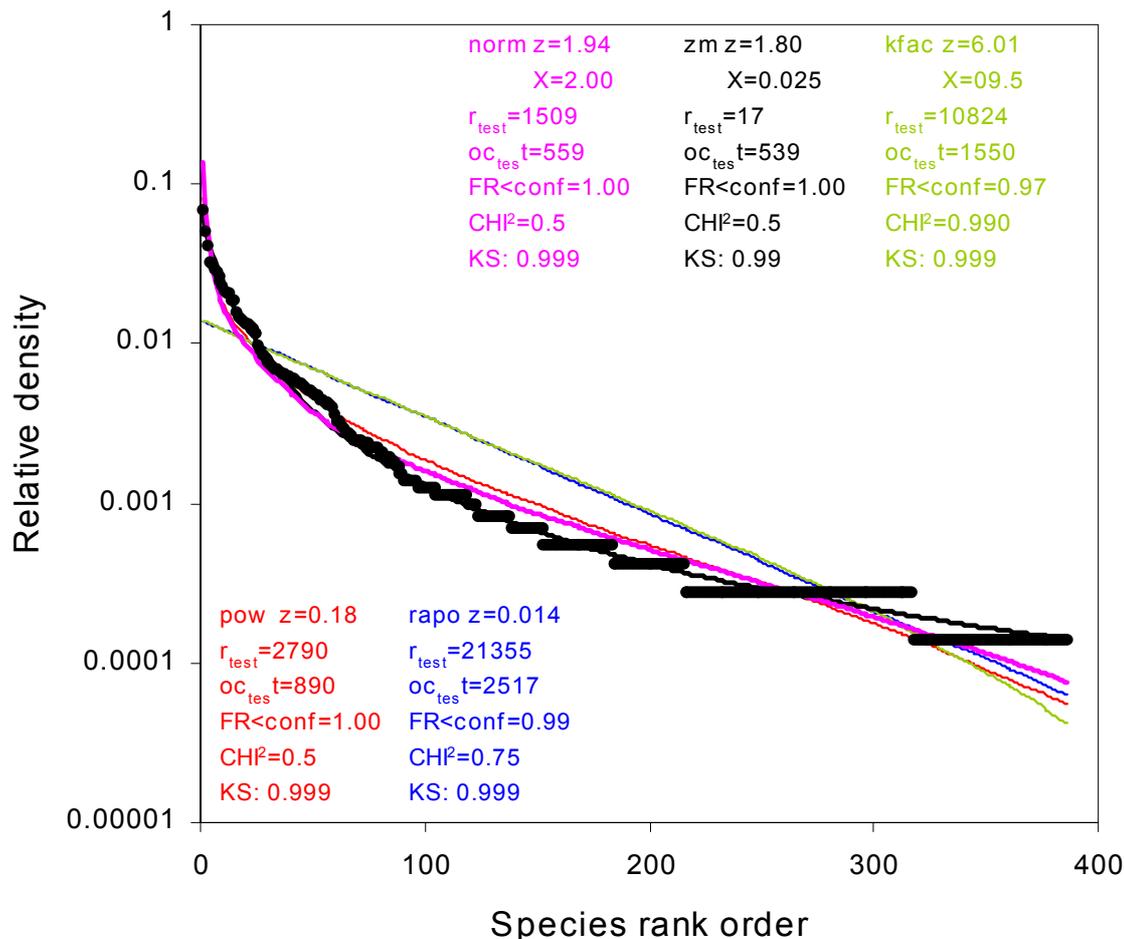


Figure 11: An example of a data fit by RAD. A sample of 386 species of Hymenoptera sampled in a beech forest on limestone (Ulrich 2001d) is fitted by five different models. The total community contains probably about 500 species. We see that except a Zipf-Mandelbrot model neither of these models (and of all others included in RAD) gives acceptable fits to the data set.

occurrences where the density differences of data set and model species of consecutive species have the same sign. *p-value* then gives the probability of rejecting the null hypothesis of a random distribution around the mean. In the case of Tab. 6 three times the sign changed but 16 times consecutive species deviated in the same way from the model.

The probability for such a pattern is less than 1% and we have to conclude, that our model does not fit the data set although in our example of Tab. 6 r_{test} and oc_{test} have values well below 10 and optical inspection of the fit (Fig. 10) looks not bad. The 95% confidence limit method also indicates no good fit. However in this case a slightly non-realistic situation was assumed. A very high sam-

ple size (*IterAss*=100) was assumed. Most often *IterAss* has to be set to 1. And then all data set species would range well inside even the 99% confidence limit.

The next six output values of the program (50%*Slope*, 50%*Interc*, 15%*Spe*, 15%*Pro*, 15%*SpeZM*, 15%*ProZM*) are the same as in the case of model computing and are already discussed above.

For real data sets with a high variability in density the Kolmogorov-Smirnov test will often give poor results because of high numbers of species with identical or similar numbers of individuals. In these cases r_{test} and oc_{test} are better suited to compare fits by different models. However, in

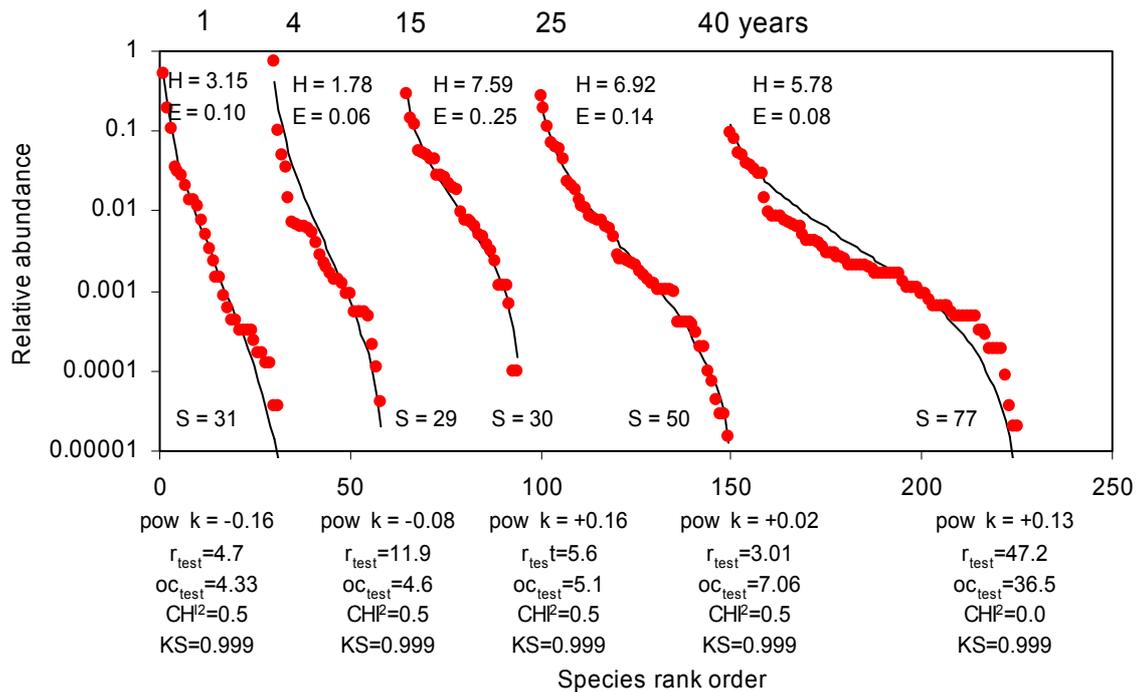


Figure 12: Another example of a data fit by RAD. Bazzaz (1975) reported 76 plant species after 40 years of old field succession. We see that that this community is much more evenly distributed than even expected from a classical broken stick model. Best fit the overlapping nich model with $z = 0.39$ and a lognormal with $z = 0.38$. H and E refer to Simpson's indices of diversity and evenness.

these cases no probability levels are available.

RAD then prints the relative abundances of the original data set and the model fitted as well as the species numbers per abundance octave of both assemblages.

At last RAD gives the progress of the fitting procedure and shows r_{test} , oc_{test} and z at each stage.

Fig. 11 contains a more complicated example of a fitting process. A sample of 386 species of forest Hymenoptera (Ulrich 2001d) - out of an estimated total of about 500 - was fitted with all models included in RAD. Five of these fits are shown in the Figure. Best fitted a Zipf-Mandelbrot model. We see that these community has more species with intermediate abundances than predicted by most models. The evenness of this non-interactive community (Ulrich 2001d) is higher than expected from classical sequential breakage or lognormal models. However, a crucial point of the

fitting process is the assumed total number of species. Enhancing this number to more than 600 favours sequential breakage or lognormal models. Unfortunately, such a high species number is not very probable (Ulrich 2001d) and we have to conclude that this community does not fit into classical assumptions about relative abundance distributions.

Another example shows Fig. 12. Bazzaz (1975) reported 77 plant species after 40 years of old field succession. This full censured community is at every stage of succession well fitted by a power fraction model (Tokeshi 1996), but not by other models, for instance a sequential breakage or a lognormal. We also see that evenness (measured by the shape generating parameter k ; Ulrich 2001b) raises during succession. Using a traditional evenness index (in this case the Simpson index) leads to a wrong impression about relative abundances.

4.4. The data set file

The input file of the fitting process can be every file in a text format, for instance 'Assemblage.txt' itself. However the first line or several lines above the beginning of the data have to start with an asterisk (*). Therefore, the file can contain an unlimited number of comment lines. RAD sorts the data after reading, you don't need to sort them prior. The number of data sets included in this file is unlimited. However, the number of command lines in the batch file must agree with the number of data sets in the input file.

RAD can fit one data set by several models or several data set by one model. Prior to fitting the program asks about this option. The input 1 fits one data set by several models, input one model to several data sets. A blank or another input will fit each data set by another model of the batch file.

RAD can read whole data matrices. The program asks prior to fit which column of the matrix to read and to fit. The default value is one. You can place the data sets either in matrix form or into one column one beneath the other. After appropriate input (value = 0 or carriage return) the program fits automatically all data sets of the matrix or the column. **If you have a data matrix, the columns have to be sorted according to species number with the species richest data set in column 1.**

4.5. Fitting Hubbel's model to a data set

The Hubbell model is individual instead of species orientated. The fitting process relies therefore mainly on the sample size (the number of individuals of the data set). If the data are given as relative abundances the user has to provide the sample size manually. Fitted assemblage and data set have therefore similar individual numbers but their species numbers will most often differ. Additionally, the model contains a second input parameter, the dispersion limitation factor that will have a strong influence on the

goodness of fit. As in the case of other two parameter models, RAD does not automatically fit for both parameters, so an estimate of this second parameter has to be provided. For larger sample sizes computing times might become extraordinary large.

5. Some problems of the program

No program is perfect. RAD sometimes has problems with certain data combinations. In these cases the program returns an error and stops. In such cases change the program settings.

Another form of error might occur during data fit. Stochastic models lead the program sometimes to run into the wrong direction. However, with some experience this type of error will immediately be encountered. A slightly change of parameter ranges forces the program nearly always to run into the correct direction.

Normally, RAD needs only a few seconds to compute a specified model. Data fitting may take longer. The computation of *hub*, *nbin* or *pow* at higher species numbers or large sample sizes, however, is sometimes time expensive. But in the case of *pow* a high species number implies a reduced variance. Therefore you can reduce CPU time by reducing the number of replicates.

6. System requirements

RAD is written in FORTRAN 95 and runs under Windows 9.x, and XP. Computation abilities are only limited by the computer's memory.

7. Citing RAD

RAD is freeware but nevertheless if you use RAD in scientific work you should cite RAD as follows:

Ulrich W. 2002 - RAD—a FORTRAN program for the study of relative abundance distributions - www.uni.torun.pl/~ulrichw

8. Acknowledgements

The development of this program was supported by a grant of the Polish Science Committee (KBN, 3 F04F 03422).

9. References

- Anscombe F. J. 1950 - Sampling theory of the negative binomial and logarithmic series distributions - *Biometrika* 37: 358-382.
- Bazzaz F. A. 1975 - Plant species diversity in old-field successional ecosystems in southern Illinois - *Ecology* 56: 485-488.
- Bell G. 2000 - The distribution of abundance in neutral communities - *Am. Nat.* 155: 606-617.
- Bersier L.-F., Sugihara G. 1997 - Species abundance patterns: the problem of testing stochastic models - *J. Anim. Ecol.* 66: 769-774.
- Fisher A. G., Corbet S. A., Williams S. A. 1943 - The relation between the number of species and the number of individuals in a random sample of an animal population - *J. Anim. Ecol.* 12: 42-58.
- Frontier S. 1985 - Diversity and structure in aquatic ecosystems (in: *Oceanography and Marine Biology - An Annual Review*, Ed. M. Barnes) - Aberdeen, pp. 253-312.
- Hubbell S. P. 1997 - A unified theory of biogeography and relative species abundance and its application to tropical rain forest and coral reefs - *Coral Reefs* 16, Suppl.: S9-S21.
- Hubbell S. P. 2001 - The unified theory of biogeography and biodiversity - Princeton (Univ. Press).
- MacArthur R. H. 1957 - On the relative abundance of bird species - *Proc. Nat. Acad. Science (US)* 43: 293-294.
- Magurran A. E. 1988 - Ecological diversity and its measurement - Princeton (Univ. Press).
- May M. 1975 - Patterns of species abundance and diversity - (in: *Ecology and evolution of communities*, Eds: M.L. Cody, J.M. Diamond) - Belknap (Cambridge), pp. 81-120.
- Motomura I. 1932 - On the statistical treatment of communities - *Zool. Mag. Tokyo* 44: 379-383 (in Japanese).
- Moulliot D., Lepretre A., Andrei-Ruiz M.-C., Viale D. 2000 - The fractal model: an new model to describe the species accumulation process and relative abundance distribution (RAD) - *Oikos* 90: 333-342.
- Pielou E. C. 1977 - *Mathematical Ecology* - John Wiley & Sons (New York), 385 pp.
- Preston F. W. 1962 - The canonical distribution of commonness and rarity. Part I and II - *Ecology* 43: 185-215, 410-432.
- Rosenzweig M. L. 1995 - Species diversity in space and time - Cambridge (Univ. Pres.).
- Sugihara G. 1980 - Minimal community structure: an explanation of species abundance patterns - *Am. Nat.* 116: 770-787.
- Tokeshi M. 1990 - Niche apportionment or random assortment: species abundance patterns revisited - *J. Animal Ecol.* 59: 1129-1146.
- Tokeshi M. 1993 - Species abundance patterns and community structure - *Adv. Ecol. Res.* 24: 111-186.
- Tokeshi M. 1996 - Power fraction: a new explanation of relative abundance patterns in species-rich assemblages - *Oikos* 75: 543-550.
- Ulrich W. 2001a - Models of relative abundance distributions I: model fitting by stochastic models - *Pol. J. Ecol.* 49: 145-157.
- Ulrich W. 2001b - Models of relative abundance distributions II: diversity and evenness - statistics - *Pol. J. Ecol.* 49: 159-175.
- Ulrich W. 2001c - Relative abundance distributions of species: The need to have a new look at them - *Pol. J. Ecol.*: 393-407.
- Ulrich W. 2001d - Hymenopteren in einem Kalkbuchenwald: Eine Modellgruppe zur Untersuchung von Tiergemeinschaften und ökologischen Raum-Zeit-Mustern - *Schriftenr. Forschenzentr. Waldökosysteme A* 171. Göttingen, 249 S.
- Ulrich W. 2002 - Community Ecology - Patterns of diversity at different scales - Toruń (Univ. Press).
- Wilson J. B. 1991 - Methods for fitting dominance/diversity curves - *J. Veg. Sci.* 2: 35-46.