

Albert - Ludwigs Universität Freiburg
Fakultät für Umwelt und Natürliche Ressourcen
Professur für Biometrie und Umweltsystemanalyse

Bachelor Thesis:

Identifying Coexistence Mechanisms from time series data

Supervisor: Prof. Dr. Carsten F. Dormann

Second Supervisor: Dr. Gesine Pufal

Christian Buschbeck

Matriculation No.: 3541464

Freiburg im Breisgau

Abstract

In ecology, coexistence mechanisms play a major part in explaining the maintenance of biodiversity. Many previous studies have examined the different mechanisms and their properties with the help of analytical models and simulations. While these studies concentrate on understanding one mechanism in detail, here the focus lays on the distinction between different mechanisms. The idea is to identify every one of them by looking at their times series data. Hence, time series were created, with a time discrete two species model. Then I suggest two different approaches to distinguish the mechanisms based on characteristic patterns that occur in the simulated population dynamics. For the differentiation, I introduce five assessment tools that quantify the observed patterns. The validation of this differentiation showed that it is possible, to significantly distinguish the mechanisms for fixed parameters. The combinations of coexistence mechanisms could be identified less significantly and not for all combinations. For alternating parameters, the significance was reduced as well. Even though, the assessment tools might still find use in indicating coexistence mechanisms and should be tested on field data. Further research needs to be done to examine the combinations of coexistence mechanisms.

Acknowledgements

I would like to thank everyone who supported me while writing this thesis.

The first to mention is Florian Hartig, who always found time to check on my progress and help me out when problems occurred.

I also thank Carsten Dormann for supervising this thesis.

Lastly, I thank my flatmate for bringing me back down to ecological thinking when my mind was stuck in modelling, and my parents for the opportunity to study at all.

Contents

Abstract

Acknowledgements

Contents

List of Figures

List of Tables

- 1. Introduction** 1
 - 1.1 Coexistence..... 1
 - 1.2 Coexistence mechanisms 2
 - 1.2.1 Classical niche partitioning..... 2
 - 1.2.2 Storage effect 3
 - 1.2.3 Relative nonlinearity of competition 4
 - 1.2.4 Neutrality..... 4
 - 1.3 Problem 4
 - 1.4 Aim of this study..... 5

- 2. Methods** 5
 - 2.1 Model description 5
 - 2.2 Simulation..... 8
 - 2.3 Assessment..... 9
 - 2.3.1 Exclusion approach..... 10
 - 2.3.2 Tools 11
 - 2.3.2 - 1 Fluctuation imbalance 11
 - 2.3.2 - 2 Fluctuation differences..... 12
 - 2.3.2 - 3 Correlation..... 13
 - 2.3.2 - 4 Mean analysis 14
 - 2.3.2 - 5 Oscillation frequency..... 15
 - 2.3.3 Classifier 16

- 3. Results** 17
 - 3.1 Confirmation of stabilization 17
 - 3.2 Classification via exclusion approach 18
 - 3.2.1 Alternating parameters 19
 - 3.3 Classification via classifier 20

4. Discussion	22
4.1 Main questions at this study	22
4.2 Answers provided by this study.....	22
4.3 Limitations	23
4.3.1 Parameters that were not alternated	23
4.3.2 Sensitivity of RNC towards environmental fluctuations	23
4.3.3 Distinguishing storage effect and combination 3	25
5. Conclusion	26
References	27
Appendix	28

List of figures

Figure 1: Relative nonlinearity	4
Figure 2: Exclusion approach scheme	10
Figure 3: Illustration of fluctuation imbalance.....	11
Figure 4: Illustration of fluctuation difference.....	12
Figure 5: Correlation between species one and two	13
Figure 6: Illustration of mean analysis	14
Figure 7: Illustration of fluctuation frequency	15
Figure 8: Matplot of the classifier matrix.....	16
Figure 9: Time to extinction	17
Figure 10: Plot of the relationship between correlation and carrying capacity	20
Figure 11: RNC in dependency to $S_d(E)$	24
Figure 12: Illustration of similarity between storage effect and combination3	25
Figure A: Population dynamics of coexistence mechanisms	28
Figure B: Population dynamics of combinations.....	29

List of Tables

Table 1: Possible parameter combinations for the model.....	8
Table 2: Parameter values for the simulation.....	9
Table 3: Time to extinction	17
Table 4: Validating the threshold values for coexistence mechanisms	18
Table 5: Validating the threshold values for combinations	19
Table 6: Accuracy of the exclusion approach.....	21
Table 7: Accuracy of the binomial classifier.....	22

1. Introduction

1.1 Coexistence

The knowledge of how species may coexist gives insight in processes of species diversity maintenance and thus how biodiversity is even possible. Because this is a major question in ecology, over the last decades a lot of research was done to get the principles of coexistence in order and to quantify them.

The most general classification of coexistence is stability: Coexistence can be stable or unstable. Unstable coexistence has a certain tendency and usually covers a small span of time until one species gets excluded. Therefore it is rather unimportant considering the question of species diversity in ecology. Stable coexistence has no tendency and covers large timespans due to the capability of species facing extinction (low density), to recover (CHESSON 2000). Whenever referred to coexistence in this study, it actually means stable coexistence because of the non-relevance of unstable coexistence.

A basic requirement for stable coexistence of two competing species is that both are able to recover from low density (which also means that each species could invade a community dominated by the other species). This criterion is referred to as *invasibility criterion* (TURELLI 1981) and is fulfilled if the geometric-mean growth rate of each species is greater than 1 in a community dominated by the other species (CHESSON 1989).

In general, coexistence mechanisms can have equalizing and/or stabilizing effects on communities. Equalizing effects lower the fitness differences between the species. Species with similar fitness are less able to competitive exclude each other. Stabilizing effects enable species facing extinction to have a positive per capita growth rate (CHESSON 2000). This grants the species with low population density (the *invader*) the possibility to recover. Equalizing effects alone can only slow down competitive exclusion. Stabilizing effects are necessary to fulfill the invasibility criterion and thus allow coexistence. Usually the invasibility criterion is fulfilled if intraspecific competition is lower than interspecific competition for a species with low population density.

Holt (2001) suggested three different classes of conditions promoting coexistence:

1. Species may coexist in a closed, temporally constant world if they experience different limiting factors at the spatial scale of the local community; this includes classical niche partitioning of resources, as well as mechanisms involving predation and parasitism, and direct interference
2. Species may coexist, even though they experience the same limiting factor, if the environment is temporally variable and species respond differently to this temporal variation (temporal niche partitioning)

3. Species may coexist if the environment is spatially open; this includes spatial niche partitioning at scales broader than local community and mechanisms such as colonization-competition tradeoffs in meta-populations

If we look at a spatially closed environment the third class gets irrelevant. As there are mechanisms which not only make coexistence possible in a variable environment (class 2) but need it to function (e.g. storage effect, nonlinearity of competition (RNC)) we may as well refer to them as “*fluctuation dependent mechanisms*” (CHESSON 2000). Therefore mechanisms of the first class can be referred to as “*fluctuation independent*”. Every one of these mechanisms is well understood and has been quantified in models (CHESSON 1992).

1.2 Coexistence mechanisms

1.2.1 Classical niche partitioning

In ecology, there are two different definitions for the term “niche” (LEIBOLD 1995):

1. For Hutchinson (1957, 1978) and Grinnell (1917) the niche is largely defined by the environmental requirements of a species (According to Leibold, Hutchinson defined his “fundamental niche”, “as a multi-dimensional "hyper-volume" describing the conditions where an organism's expected absolute fitness is at least zero, in a conceptual space whose axes include all of the environmental variables affecting that species.”).
2. Elton (1927) and Mac Arthur and Levin (1967) rather argued the impact a species has on its environment is defining its niche.

Leibold (1995) considered aspects of both criteria important and suggested the use of the terms “requirement niche”, “impact niche” and “total niche”.

Coexistence is possible if (at least at low density) intraspecific competition is greater than interspecific competition (CHESSON 2000). This can be accomplished, by increasing the difference between the species’ niches. If a species evolves differences due to natural selection, its’ niche is changing. This process is called niche partitioning. Even if species have an overlap in one of the three general resource dimensions (food type, habitat, time) coexistence is possible because this similarity may not be given in a second resource dimension (e.g. Food type and time: Species feeding from the same insects at different times), even if the two dimensions are the same (e.g. Habitat and habitat: vertical habitat may differ with similar horizontal habitat). This mechanism is termed resource Partitioning (SCHOENER 1974).

1.2.2 Storage effect

Simply put, in a variable environment species are able to “store” benefits from time when conditions are in their favor, to endure times of unfavorable conditions due to the storage effect. The effect is a combination of three components. Fundamental is the assumption, that different species that experience the same environmental changes respond differently to these changes (“*differential responses to the environment*” (CHESSON 2000)). That alone would not lead to coexistence because the environment does not depend on population densities. But it does have an effect on the competition between species. In fact, good environment leads to high competition, while bad environment leads to low competition. This connection is termed “*covariance between environment and competition*” (CHESSON 2000). As species respond different to their environment it is possible that the status quo is favorable for species A and unfavorable for species B. In this case species A is experiencing mostly intraspecific competition and species B mostly interspecific competition. To prevent species B from extinction the third component, “*buffered population growth*” (CHESSON 2000), is necessary. Lots of species have Life-history-stages that have a buffering effect (e.g. seed banks, resting eggs, long lived adults). Due to these stages they are able to maintain a high recruitment rate despite unfavorable conditions. By doing so, they diminish interspecific competition compared to intraspecific competition, which is stabilizing. Besides Life-history-stages there are other ways to buffer population growth (CHESSON 2000).

1.2.3 Relative nonlinearity of competition (RNC)

The underlying principle for RNC is a nonlinear relationship between feeding rates and resource availability of competing species (HOLT 2001). Chesson phrased it as nonlinear relationship between recruitment rate and the magnitude of a common limiting factor (CHESSON 2000). A Species that is competitive superior (higher per capita recruitment-/feeding- rate) when the impact of a common limiting factor is low, may be inferior while it is high. For the competing species the situation is vice versa (shown in figure 1). Thus, there are conditions benefiting the first species and conditions benefiting the second (HOLT 2001). Here, one can obviously see the fluctuation dependence: In an environment without fluctuation the species which benefits from the current situation wins. Due to the variability in the environment both species experience times of competitive superiority and times of inferiority. Because environmental variability is not density dependent, even a population facing extinction is able to become superior and recover from low density. Hence RNC has a stabilizing property and promotes coexistence.

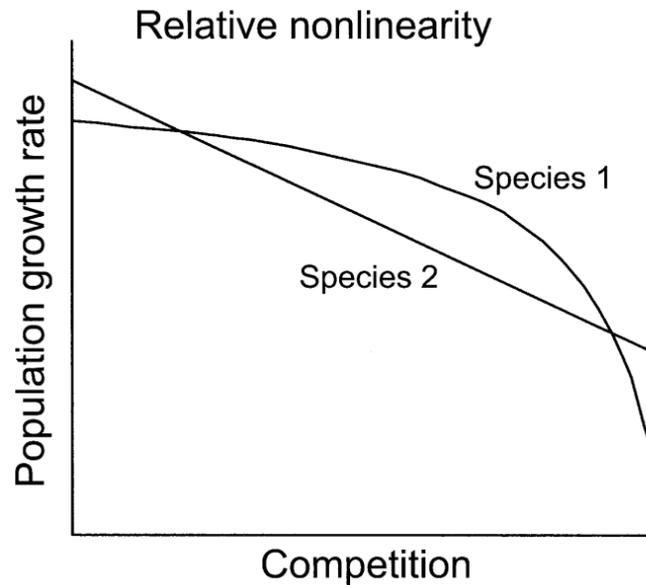


Figure 1 (ROXBURGH et al. 2004): Illustrates the nonlinear relationship between the strategies two competing species have on dealing with a limiting factor (here phrased as competition). For very low or very high competition, species 2 shows the higher population growth rate (compared to the competing species), while species 1 has higher population growth for intermediate competition.

1.2.4 Neutrality

Neutrality is not a coexistence mechanism. In fact Hubbell's Neutral Theory of Biodiversity (HUBBELL 2001) rather argues that a lot of species may not be coexisting in the way defined above, but are on "a slow random walk to extinction" (SOMMER 2002). All the previous described mechanisms promote coexistence mainly due to stabilizing properties. In contrast, the random walks in neutral communities occur because of strong equalizing effects. High equality means the species fitness do not differ very much and therefore they have similar per capita reproduction-, death- and immigration rates. Only demographic and environmental stochasticity leads to fluctuating population densities, resulting in the mentioned random walk. However, because it is possible for neutral communities to persist for long periods of time without extinction it is a relevant alternative to coexistence in the mathematical sense (SOMMER 2002) and therefore needs to be considered here.

1.3 Problem

Models help us to simulate population dynamics or to get information, whether coexistence is possible or not. However, little is known how these mechanisms work together and to which degree each of them influences the population dynamics. Chesson introduced a tool to assess coexistence mechanisms relying on recruitment fluctuations. For example in an invader-residence scenario, where both species

have similar sensitivity to competition, the difference between invader and resident recruitment variance is proportional to the standard quantitative measure of the storage effect for an individual invader, and the average invader variance compared with average resident variance is proportional to the community average storage effect (CHESSON 2003). But still predictions can only be made for certain assumptions and not always clearly. To distinguish which mechanism is responsible for coexistence with given population dynamics is not possible yet. Most previous work analyzed the models in a mechanistic way. Less effort was made to examine them graphically. Therefore, tools for analyzing time series are necessary. Once validated, these tools could be used in the field to help ecologists gain knowledge about the circumstances under which their objects of study coexist.

1.4 Aim of this study

In order to gain knowledge about the contribution coexistence mechanisms have on population dynamics, it is necessary to identify specific population dynamic patterns for each of them. That is why I created time series for the following coexistence mechanisms: niche partitioning (NP), relative nonlinearity of competition (RNC), storage effect (SE) and neutral dynamics. Time series were created for combinations of these mechanisms as well. To provide comparability they were all simulated with the same model but altering input parameters. For same reasons, demographic and environmental fluctuations have been considered, even for fluctuation independent mechanisms. The resulting time series have been analyzed and I suggest two different approaches to identify the underlying mechanism(s).

2 Methods

2.1 Model description

In order to create time series of population dynamics I created a 2-species population model with which it was possible to switch on and off the desired mechanism by setting the right parameters. The foundation was the Maynard Smith and Slatkin model (MSS) which models reproduction ratios for the case of **relative nonlinearity of competition**:

$$f(N) = \frac{r}{1+(r-1)*\left(\frac{N}{K}\right)^b} \quad (1)$$

Where N is the population size and f represents the ratio of N(t+1) and N(t). The intrinsic growth rate is given by r. K is the carrying capacity. The parameter b reflects the density compensation strategy

Methods

(HARTIG et al. 2014). High b-values lead a species to overcompensate and hence experience strong fluctuations. If the other species has a different density compensation strategy (small b value) coexistence is possible because the compensator is able to grow while the over compensator has a setback. The mechanism of **niche partitioning** is easily modeled via the Lotka-Volterra equations. If species occupy different niches, they reduce the competitive impact they have on each other. In the Lotka-Volterra equation this effect is represented by a factor α . I added a term of Lotka-Volterra Competition to the population-capacity ratio, which was done before by Münkemüller et al. (2009). Resulting in the following formulas for species one and two:

$$f_1(N) = \frac{r_1}{1+(r_1-1)*\left(\frac{N_1+\alpha_{12}*N_2}{K_1}\right)^{b_1}} \quad (2)$$

$$f_2(N) = \frac{r_2}{1+(r_2-1)*\left(\frac{N_2+\alpha_{21}*N_1}{K_2}\right)^{b_2}} \quad (3)$$

With α_{12} being the competitive effect species two has on species one and vice versa for α_{21} . For alpha values below one, coexistence is promoted. If α_{12} and α_{21} are set to one, both species will experience **neutral dynamics** (CARROLL et al. 2015). The storage effect needs environmental fluctuation in order to function. So I added an additional term (E) to simulate the environmental response for each species.

$$f_1(N) = \frac{*r_1}{1+(r_1-1)*\left(\frac{N_1+\alpha_{12}*N_2}{E_1*K_1}\right)^{b_1}} \quad (4)$$

$$f_2(N) = \frac{r_2}{1+(r_2-1)*\left(\frac{N_2+\alpha_{21}*N_1}{E_2*K_2}\right)^{b_2}} \quad (5)$$

In the model the environmental response (E) affects the carrying capacity K. This makes sense because in good years one species' habitat can carry more individuals than in bad years. The values for (E) originate from a log-normal distribution. In case the storage effect is effective following term ensures both species have *different environmental responses*:

$$E_1 = \text{lognorm}(x) \quad (6)$$

$$E_2 = 1/E_1 \quad (7)$$

For all cases where the storage effect is not involved we can simply multiply the reproduction ratio with the current population density to get the population density for the next time step. Every time step adults die. In the model this is regarded by multiplying the population density for the next time step with the difference of 1 and the dying rate (d).

Methods

$$N_1(t + 1) = N_1(t) * (1 - d) * f_1(N) \quad (8)$$

$$N_2(t + 1) = N_2(t) * (1 - d) * f_2(N) \quad (9)$$

The **storage effect** is modeled a little different. One key ingredient for the storage effect is a buffered population growth. Here, life history stages enable species to lower the impact competition has on them. To represent these life history stages (e.g. a seed bank) I modeled a buffer (B). At first, the whole reproduction of one species fills up the buffer by adding it to the current buffer value. If we think of B as a seed bank, it makes sense to multiply the input for the buffer by 1000, because the production of seeds is much greater than the actual reproduction of individuals.

$$B_1(t) = B_1(t) + N(t) * f_1(N) * 1000 \quad (10)$$

$$B_2(t) = B_2(t) + N(t) * f_2(N) * 1000 \quad (11)$$

As said before only a small part of the seeds within the seed bank will germinate. This rate of germination also depends on the current environmental conditions. The germinated seeds represent the population density of the next time step.

$$N_1(t + 1) = 0.00005 * B_1(t) * E_1 \quad (12)$$

$$N_2(t + 1) = 0.00005 * B_2(t) * E_2 \quad (13)$$

The germinated seeds are not in the seed bank (B) anymore and need to be subtracted. Some seeds are eaten by animals. Others might rot or are destroyed otherwise. That's why a certain portion of the buffer is subtracted each time step.

$$B_1(t + 1) = B_1(t) - N_1(t + 1) - 0.2 * B_1(t) \quad (14)$$

$$B_2(t + 1) = B_2(t) - N_2(t + 1) - 0.2 * B_2(t) \quad (15)$$

To take demographic stochasticity into account the resulting values for $N(t+1)$ are drawn from a poisson-distribution.

To sum up how the model can represent the different mechanisms Table 1 shows the specific parameter combinations that are required for the mechanisms and their combinations (buffered population growth cannot be influenced by the parameter combination. It is turned on or if via an if-loop). The interval for possible b_1 - and b_2 -values could be metered from the invasibility plot in Hartig et al. (2014). Although we are looking at 4 mechanisms there are only 4 possibilities to combine them.

That is, because neutrality rather is the absence of any mechanism at all. As soon as it is combined with one of the others, neutrality is lost.

Table 1: Possible parameter combinations for the model

Mechanism	Parameter
Neutral Dynamics	$\alpha_{12} = \alpha_{21}$
RNC	$b_1 \in \{x \mid 0.4 < x < 1\}$ $\wedge b_2 \in \{x \mid 3 < x < 8\}$ $\vee b_1 \in \{x \mid 3 < x < 8\}$ $\wedge b_2 \in \{x \mid 0.4 < x < 1\}$
Niche Partitioning	$\alpha_{12}, \alpha_{21} < 1$
Storage Effect	$sd(E) > 1.2$ buffer on (se=1)
Combination	Parameter
Combination1 (RNC and SE)	$b_1 \in \{x \mid 0.4 < x < 1\}$ $\wedge b_2 \in \{x \mid 3 < x < 8\}$ $\vee b_1 \in \{x \mid 3 < x < 8\}$ $\wedge b_2 \in \{x \mid 0.4 < x < 1\}$ $sd(E) > 1.2$ buffer on (se=1)
Combination2(RNC and NP)	$b_1 \in \{x \mid 0.4 < x < 1\}$ $\wedge b_2 \in \{x \mid 3 < x < 8\}$ $\vee b_1 \in \{x \mid 3 < x < 8\}$ $\wedge b_2 \in \{x \mid 0.4 < x < 1\}$ $\alpha_{12}, \alpha_{21} < 1$
Combination3(SE and NP)	$\alpha_{12}, \alpha_{21} < 1$ $sd(E) > 1.2$ buffer on (se=1)
All	$b_1 \in \{x \mid 0.4 < x < 1\}$ $\wedge b_2 \in \{x \mid 3 < x < 8\}$ $\vee b_1 \in \{x \mid 3 < x < 8\}$ $\wedge b_2 \in \{x \mid 0.4 < x < 1\}$ $\alpha_{12}, \alpha_{21} < 1$ $sd(E) > 1.2$ buffer on (se=1)

2.2 Simulation

The model described above was used to create time series for the different coexistence mechanisms in R Studio. In total, eight time series were created by using mechanism-specific parameter combinations for each mechanism and combinations of them (Table 1). Population dynamic plots are given in the appendix (A and B). Despite parameters that influence the coexistence mechanism, there are other input variables. To maintain comparability for each simulation these parameters were set for a fix value, given in Table 2. For the same reason, environmental stochasticity was added to every simulation. However it was not possible to run the simulations with the same magnitude of

Methods

fluctuations (standard deviance (E)). That is, because the storage effect needs strong fluctuations in order to function while, for example RNC cannot compensate strong fluctuations, leading to the extinction of one species (details in discussion). Thus, every simulation with storage effect involved was run with higher sd(E)-Values than the ones without (Table 2).

Table 2: Parameter values for the simulation

Parameter	$N_1(1), N_2(1)$	r_1, r_2	K_1, K_2	α_{12}	α_{21}	b1	b2	d_1, d_2	t_{\max}	sd(E)
Value	200	5	500	1	1	1	1	0.05	500	0.15
Value (Mechanism)	-	-	-	0.6 (NP)	0.9 (NP)	5 (RNC)	-	-	-	1.2 (SE)

2.3 Assessment

Each time series was examined to find characteristic patterns. These patterns were quantified by using different assessment tools that I will introduce below. The application of these tools follows a certain schema (Figure 2), with which it is possible to determine the effective coexistence mechanism(s) for unknown population dynamics. However, I failed in finding an assessment tool that can significantly distinguish the Storage effect from Combination 3 (SE and NP). Therefore it was also not possible to distinguish the combination with all 3 mechanisms involved. A detailed description of the problem is given in the discussion. None of the tools uses absolute values. In fact, ratios are used in order to evade trivial solutions for the differentiation.

2.3.1 Exclusion approach

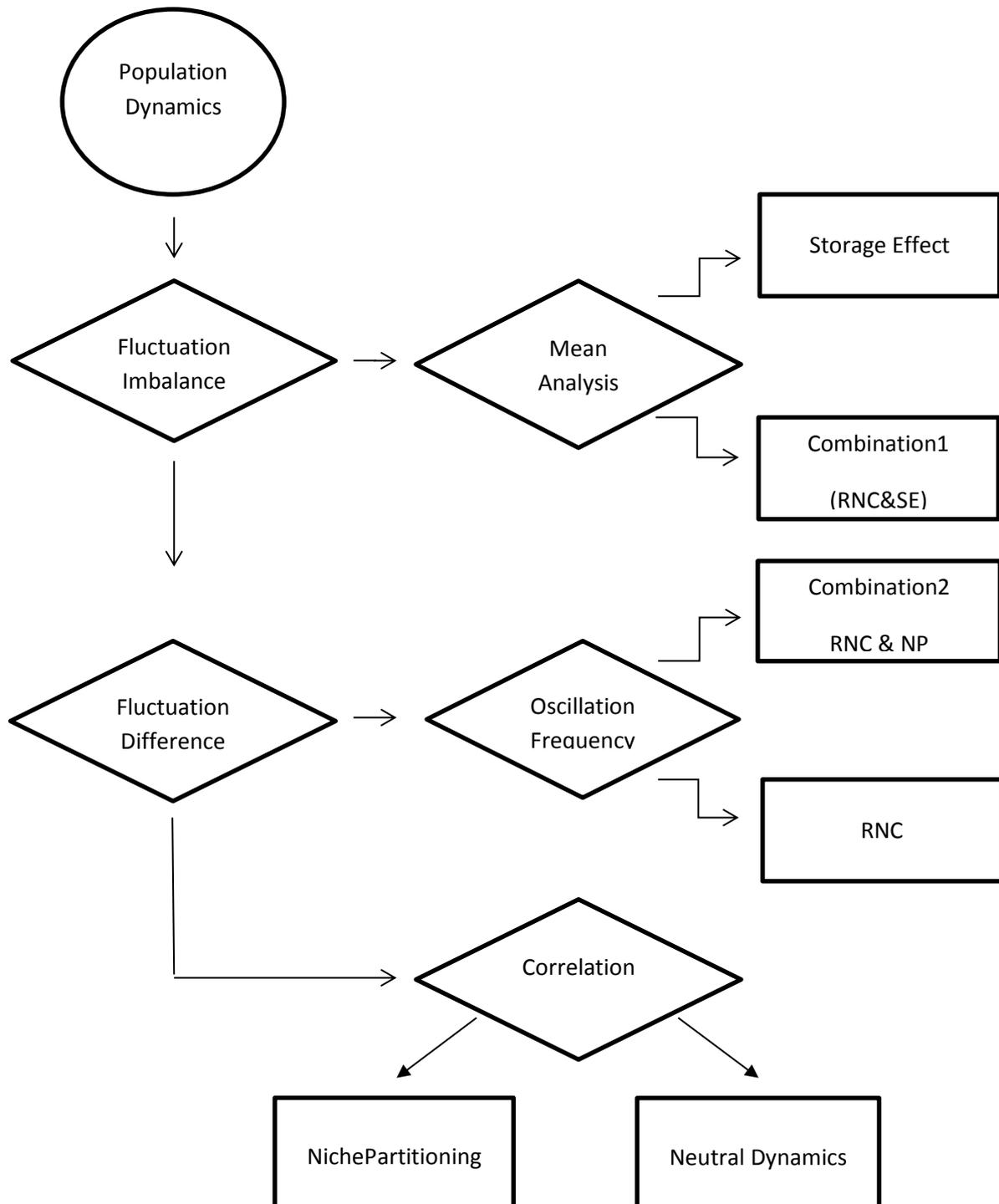


Figure 2: This flowchart shows the procedure of the Exclusion Approach. The rhombs represent the 5 assessment tools while the rectangles represent the outcome (i.e. the determined mechanisms). The circle stands for the time series which is to be tested.

2.3.2 Tools

1. Fluctuation imbalance (identifying storage effect)

The population dynamics induced by the storage effect show a tendency towards great fluctuations above the mean population density, but not below (Figure 3). As a result of these *imbalanced fluctuations*, the number of population density-values below the mean, are greater than above. To quantify this effect I suggest following formula.

$$I = \frac{n_{1b}}{n_{1a}} * \frac{n_{2b}}{n_{2a}} \quad (16)$$

Where n_b is the number of population density values below the mean. Respectively n_a is the number of population density values above the mean.

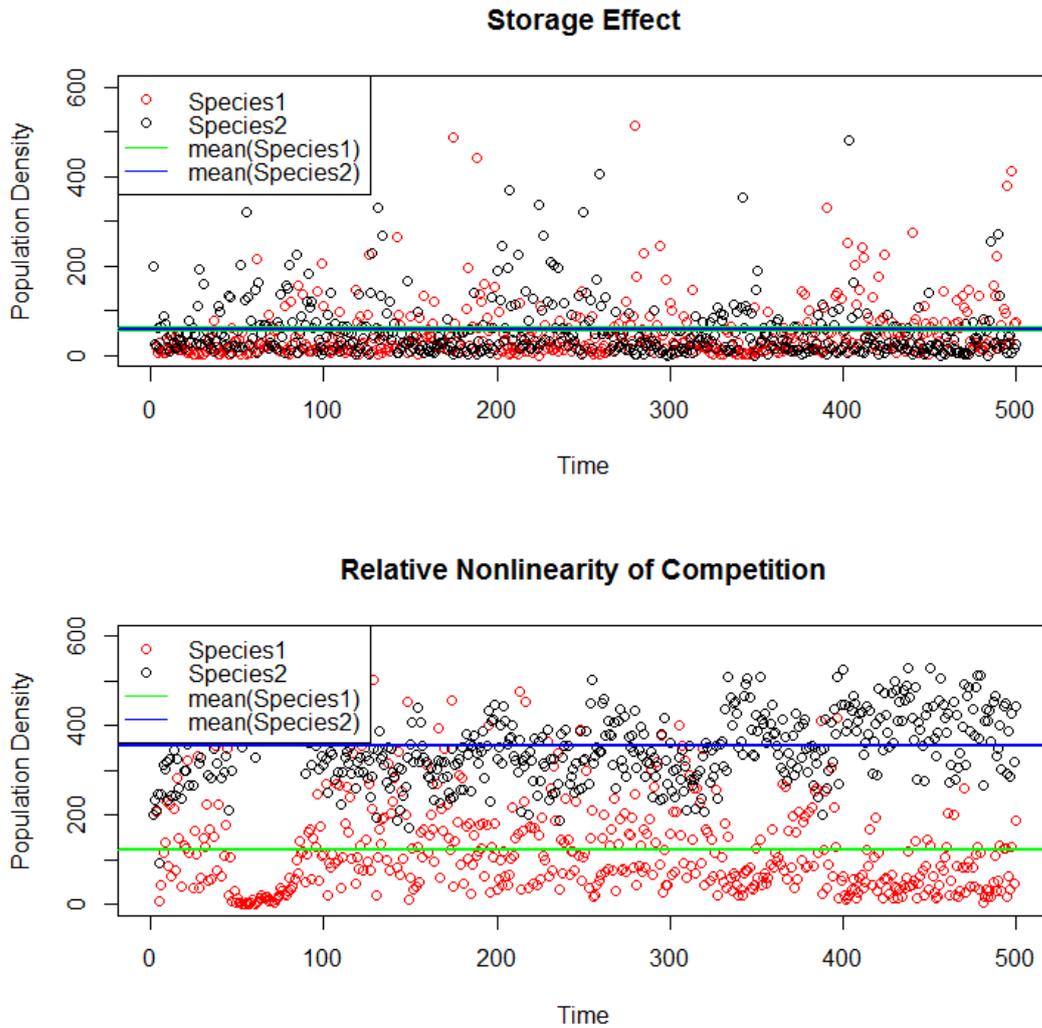


Figure 3: The population dynamics induced by the storage effect show that most of the population density Values are below the mean. In comparison, in the RNC’s dynamics the population density fluctuate above and below the mean in an equal manner.

2. Fluctuation differences (identifying RNC)

The most obvious pattern of population dynamics induced by RNC is the difference in magnitude of fluctuations between the competing species (Figure 4). While the overcompensating species oscillates with large amplitudes, the compensating species experiences moderate fluctuations. In order to quantify these *fluctuation differences* I suggest following formula:

$$D = \left| \frac{\sum_{t=1}^{t_{max}} \left(\frac{|N_1(t) - N_1(t-1)|}{N_1(t)} \right)}{t_{max}} - \frac{\sum_{t=1}^{t_{max}} \left(\frac{|N_2(t) - N_2(t-1)|}{N_2(t)} \right)}{t_{max}} \right| \quad (17)$$

With t_{max} being the total number of time steps.

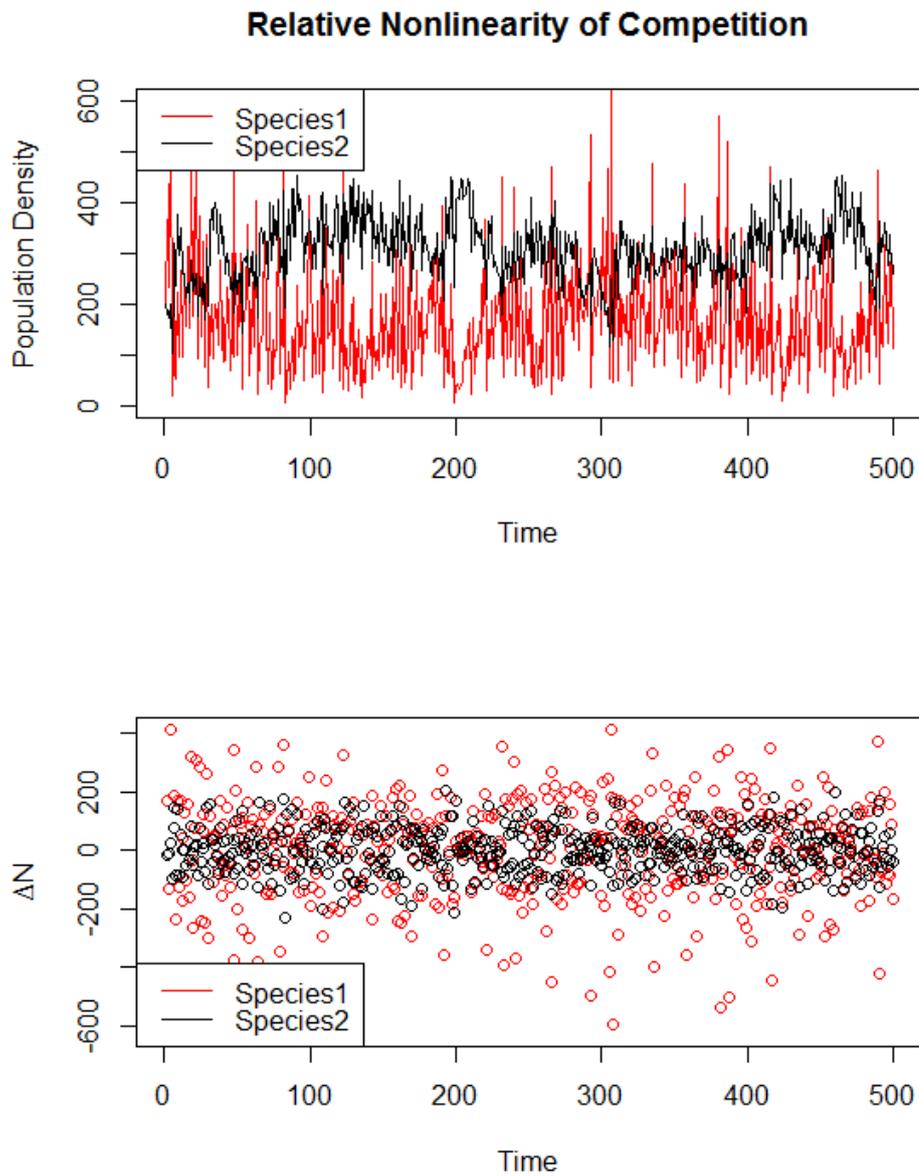


Figure 4: Species one’s fluctuation amplitude is much greater than those of species two. The plot $\Delta N \sim \text{Time}$ illustrates this phenomenon.

3. Correlation between species one and two (distinguishing neutrality from niche partitioning)

Neutral population dynamics exhibit different correlations between population densities of the competing species than the ones induced by niche partitioning (Figure 5). That's why Pearson's correlation coefficient r can be used to tell them apart.

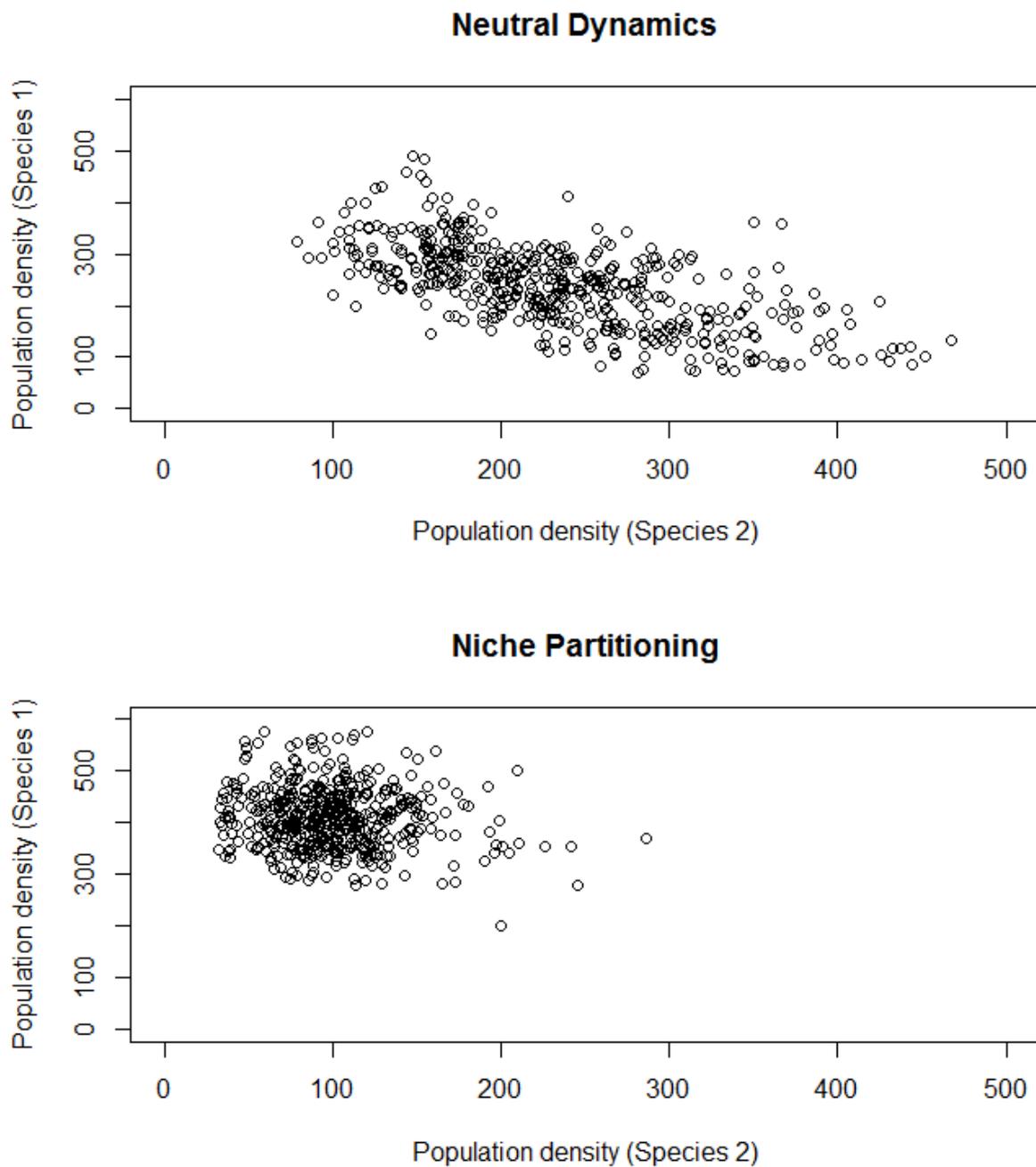


Figure 5: The population densities of species one and two are plotted against each other. It is visible that for neutral dynamics the correlation is higher (pearsons' $r = -0.65$) than for niche partitioning ($r = -0.14$)

4. Mean analysis (identifying Combination 1)

If we compare the population dynamics of Combination 1 to the storage effect it is easy to detect the influence RNC has on the dynamics (Figure 6). While in the storage effects' time series both species experience similar fluctuations there is a difference between them in Combination1. Here, one species (the one with the higher b-value), is able to grow much faster than the other, after its population crashed. This leads to one species having a higher value of mean population density compared to the competing species. Thus, I suggest following formula:

$$M = \left| 1 - \frac{mean(N_1)}{mean(N_2)} \right| \quad (18)$$

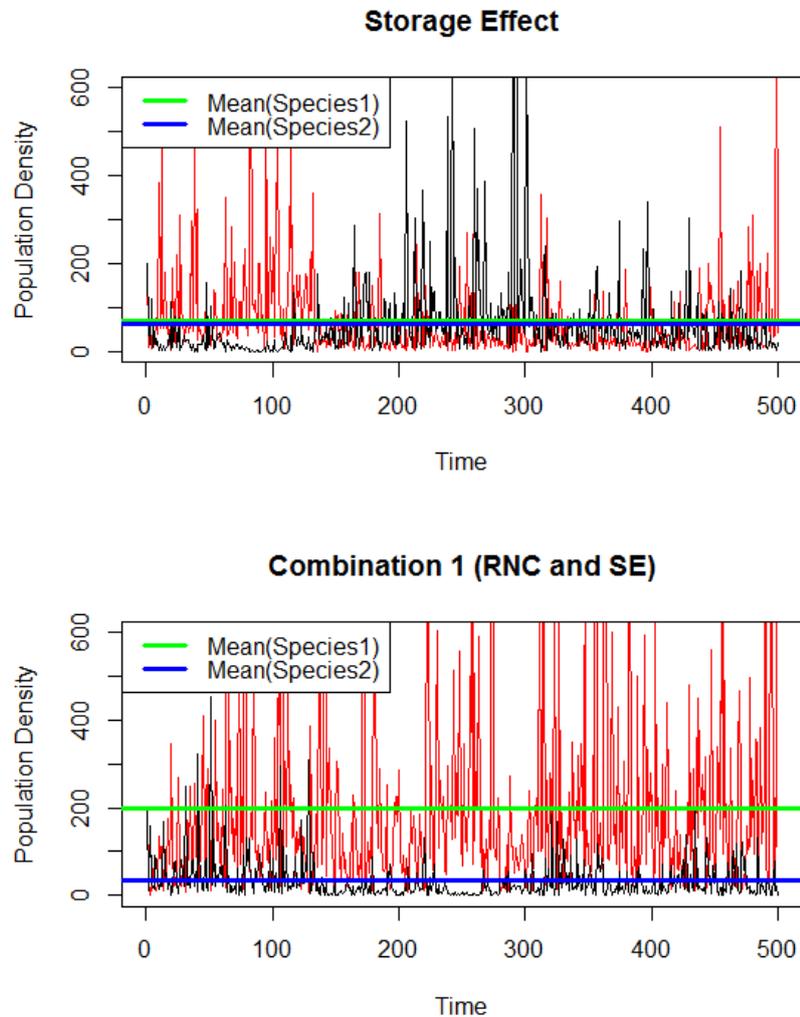


Figure 6: In Combination 1 the influence of RNC is visible because Species 1 (the overcompensator) fluctuates much more. As result, the mean of the overcompensator is always higher. In case of the storage effect the means are more or less the same.

5. Oscillation Frequency (identifying Combination 2)

Both RNC and Combination 2 create heavy fluctuations for one of the species (overcompensator). But in most of the time series created by Combination 2 strong to moderate fluctuations occur more often than in the ones created by RNC (Figure 7). To quantify this fluctuation frequency, it is possible to create a ratio between the times when the absolute fluctuations of the overcompensating species are below half of the mean population density, and the total number of time steps.

$$F = \frac{|\{|\Delta N_o| < \frac{\text{mean}(N_o)}{2}\}|}{t_{max}} \quad (19)$$

Where N_o is the population density of the overcompensating species and t_{max} the total number of time steps.

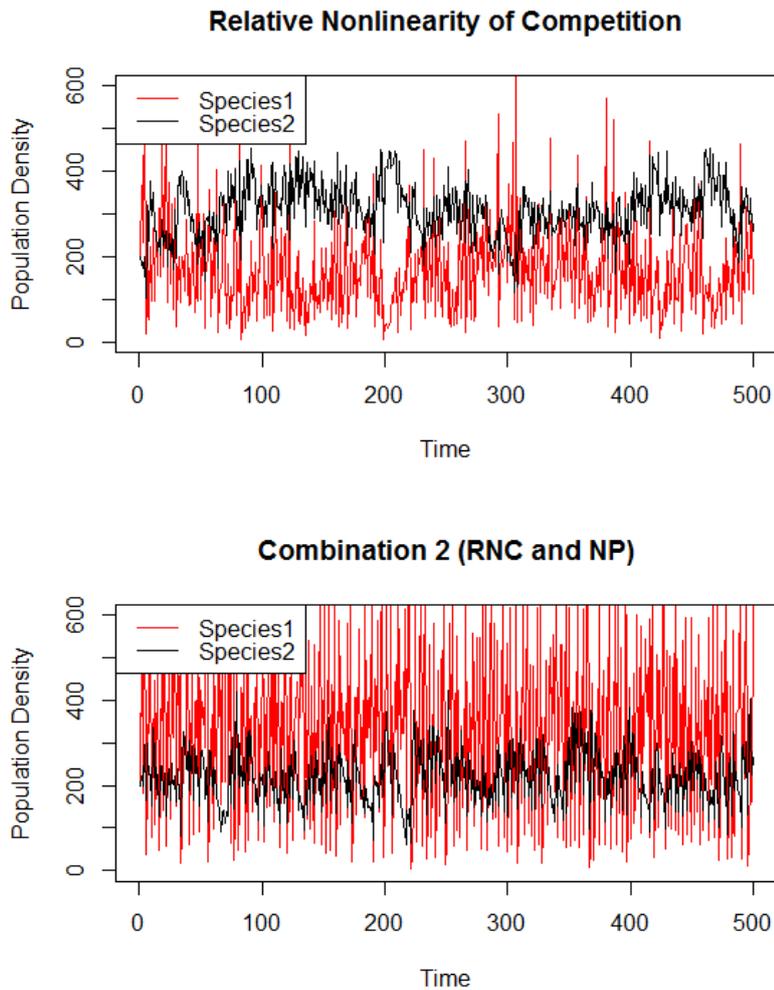


Figure 7: This plot shows the higher frequency of heavy fluctuations in the case of Combination 2, compared to RNC.

2.3.3 Classifier

Alternatively to the exclusion schema a binomial classifier (function *train* in R Studio) to distinguish the time series was used. When provided with a data set for training, this function is able to determine characteristic patterns in unknown data and assign classes according to the training data set. For identifying the four mechanisms for fixed parameters, the training data set was a matrix consisting of three predictors and one class. The model was run one hundred times for each mechanism. Each time, the first three assessment tools were calculated and stored in the matrix as predictors. Accordingly, the simulated mechanism was stored in the class column. Figure 8 shows the predictors' values in the matrix. For the classification, the model “*Conditional Inference Random Forest*” was chosen. For unknown population dynamics it is now possible to identify the mechanism by executing the fit on the unknown data. This is done by predicting the class (function *predict*), with the values for the three assessment tools for the unknown population dynamic as predictors, and the trained fit from before.

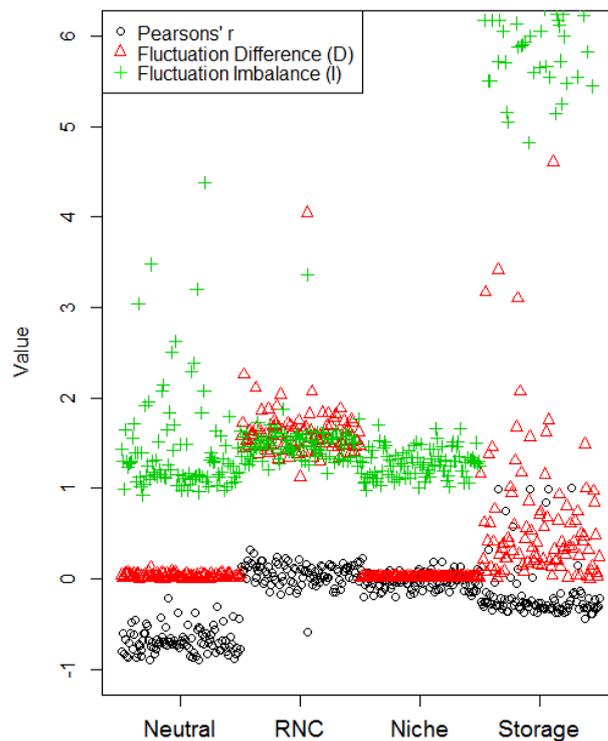


Figure 8: A matplot of the matrix used as training data set. Each mechanism has been simulated one hundred times. The values of the calculated predictors are shown here. The column containing the class was left out because of triviality. Each mechanism has been simulated one hundred times. This plot shows the characteristic patterns the mechanisms produce for the assessment tools. For example the fluctuation imbalance has much greater values in case of storage effect than for the others. This makes sense and is important, because it was designed to distinguish the storage effect from the rest. The same is true for fluctuation difference and RNC as well as pearsons' r and neutral dynamics.

3. Results

3.1 Confirmation of stabilization

In order to evaluate the stabilizing properties of a mechanism or combination I calculated the time to extinction (TTE). To do so, the model was run one hundred times for 10 000 time steps, for each mechanism and combination with fixed parameters (Table 2). In each simulation the time steps until one species' population density equals zero were stored in a vector. In the case of storage effect it sometimes occurs that one species' population density is zero but it is not extinct because its buffer is still filled. That's why in these cases an additional condition for extinction was that the buffer is below 1. The mean of the stored values are given with "TTE" in Table 3. It shows, that besides neutral dynamics every mechanism and combination provides coexistence for more or less long time spans, which indicates stabilization. Examples of these simulations are shown in Figure 9.

Table 3: Time to extinction

Mechanism	Neutral	RNC	Niche	Storage	Comb1	Comb2	Comb3	All
TTE	588	7350.8	8769.61	8014.63	8636.23	8405.19	5686.83	7104.92

TTE shows the mean time steps until one species gets extinct from 100 simulations, 10000 time steps and fixed parameters

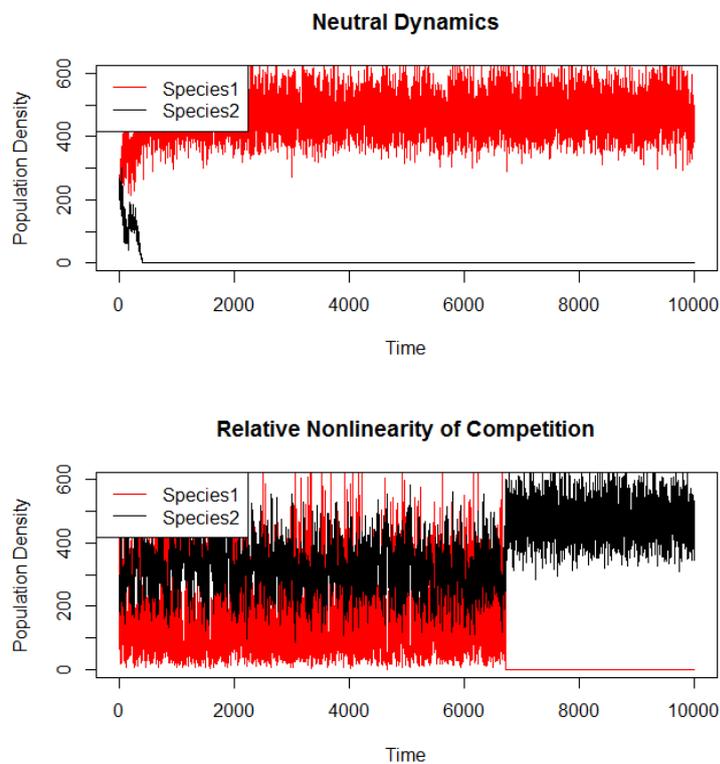


Figure 9: Exemplarily shows the time until one species gets extinct for the cases neutral dynamics (TTE = 405), and RNC (TTE = 6715)

3.2 Classification via exclusion approach

After having validated the functioning of the coexistence mechanisms and their combinations, I turn to the question of whether they can be differentiated from the created times series of two species.

As first approach I suggest to follow the exclusion schema using the first three assessment tools for the differentiation of the four mechanisms (Neutral, RNC, NP, SE). However, the tools need to be validated first in order to create reliable results.

For the validation, the formulas above were used to calculate values for the assessment tools *Fluctuation Imbalance*, *Fluctuation Difference*, and *Correlation between Species one and two*, for each mechanism, for one thousand times. The parameters were set according to Table 2. After examining the calculated values, it was possible to estimate threshold-values. In order to validate the estimates, two Exact Binomial Tests were applied on each threshold value. For the first test the number of simulations where the mechanism which should be identified was above the threshold value counted as success. For the second test the number of simulations where the remaining mechanisms were below the threshold value counted as success. As table 4 shows, for most parts the estimated values are highly significant.

Table 4: Validating the threshold values for the identification of coexistence mechanisms

Assessment Tool	Mechanism	Threshold Value	Success If	Probability of Success	Confidence Interval (95%)
Fluctuation Imbalance(I)	Storage Effect	4.5	$I > 4.5$	1.000	[0.996; 1.000]
Fluctuation Imbalance (I)	RNC, NP Neutral	4.5	$I < 4.5$	0.997	[0.994; 0.999]
Fluctuation Difference (D)	RNC	1	$D > 1$	1.000	[0.996; 1.000]
Fluctuation Difference (D)	SE, NP Neutral	1	$D < 1$	1.000	[0.996; 1.000]
Correlation(r)	Neutral	0.5	$ r > 0.5$	0.933	[0.916; 0.948]
Correlation(r)	RNC,SE,NP	0.5	$ r < 0.5$	0.995	[0.992; 0.997]

In order to test the exclusion approach, one thousand simulations were run with the simulated mechanism being chosen at random. For every simulation the used mechanism was stored for later comparison. Then the approach was applied, following the exclusion schema (Figure 2), and using the validated threshold values. The result was compared to the stored “true” mechanisms. An exact binomial test calculated a probability of success of 0.985 (Confidence Interval (95%): [0.975; 0.992]). This means for fixed parameters it is possible to significantly distinguish the four mechanisms.

Validating assessment tools (Mechanisms and Combinations)

Following the schema it is possible to distinguish Combination 1 and the storage effect by using the Mean analysis. Accordingly Combination 2 and RNC can be distinguished with the Oscillation Frequency Analysis. Therefore, threshold values for these assessment tools were estimated and validated like above. Table 5 shows the result of that validation.

Table 5: Validating the threshold values for the identification of combinations

Assessment Tool	Mechanism	Threshold Value	Success If	Probability of Success	Confidence Interval (95%)
Mean Analysis (M)	Combination1	0.90	$M > 0.9$	1.000	[0.996; 1.000]
Mean Analysis (M)	Storage Effect	0.90	$M < 0.9$	0.974	[0.962; 0.983]
Oscillation Frequency (F)	Combination 2	0.03	$F < 0.03$	0.990	[0.982; 0.995]
Oscillation Frequency (F)	RNC	0.03	$F > 0.03$	0.988	[0.979; 0.994]

The simulations were tested the same way as above. As result the differentiation of the four mechanisms and two of their combinations are a little less significant than the one without the combinations (probability of success: 0.94; Confidence Interval (95%): [0.923; 0.954]). Here the differentiation between storage effect and Combination 2 is the least powerful test and for most part responsible for the worse result (probability storage effect was misjudged for Combination 1 was 0.12 and vice versa 0.15).

3.2.1 Alternating parameters

So far the model was run with fixed parameters which made the examination much easier. However, for application it is necessary that the assessment tools are robust enough to work with alternating parameters. For the alternation, the values for the carrying capacity were drawn from a sequence in the interval [500; 10 000] for every simulation. In the case of niche partitioning, Alpha_{12} and Alpha_{21} were each drawn from sequences with the intervals [0.1, 0.9]. In the case of RNC, b_1 was drawn from either a sequence with the interval [0.4, 1.0] or a sequence with the interval [3, 7]. Respectively, b_2 was drawn from the sequence that was not chosen for b_1 so that the two species differ in their way of resource use. With increasing carrying capacity the correlation between the competing species' population density changes, in case of neutral dynamics and niche partitioning. That is why it is not sufficient to determine one absolute value as threshold. Thus, I suggest a graphical approach for estimating the threshold value in dependency to the carrying capacity. To do so I plotted the correlation of the competing species for neutral dynamics and niche partitioning for the complete carrying capacity interval [500,10000] in steps of one (Figure 10). Then I added two lines which

Results

separate most of the correlation values generated by neutral dynamics from the ones generated by niche partitioning. Via the loess function I predicted values for a smoothed curve. A root function was fitted to the predicted values, using the *nls*-function. With this fit it is possible to estimate a threshold value (r_{crit}) in dependency of K. For the analysis the K - value can be estimated by adding up the mean population densities of the competing species. Leading to following formula for r_{crit} :

$$r_{crit} = \frac{(\text{mean}(N) + \text{mean}(N))^{\frac{1}{5.2877125}}}{3.0500907} - 0.9590107 \quad (20)$$

As described above the probability of success for the identification of the mechanisms and combinations was calculated via an exact binomial test. Table 6 shows the results of these calculations.

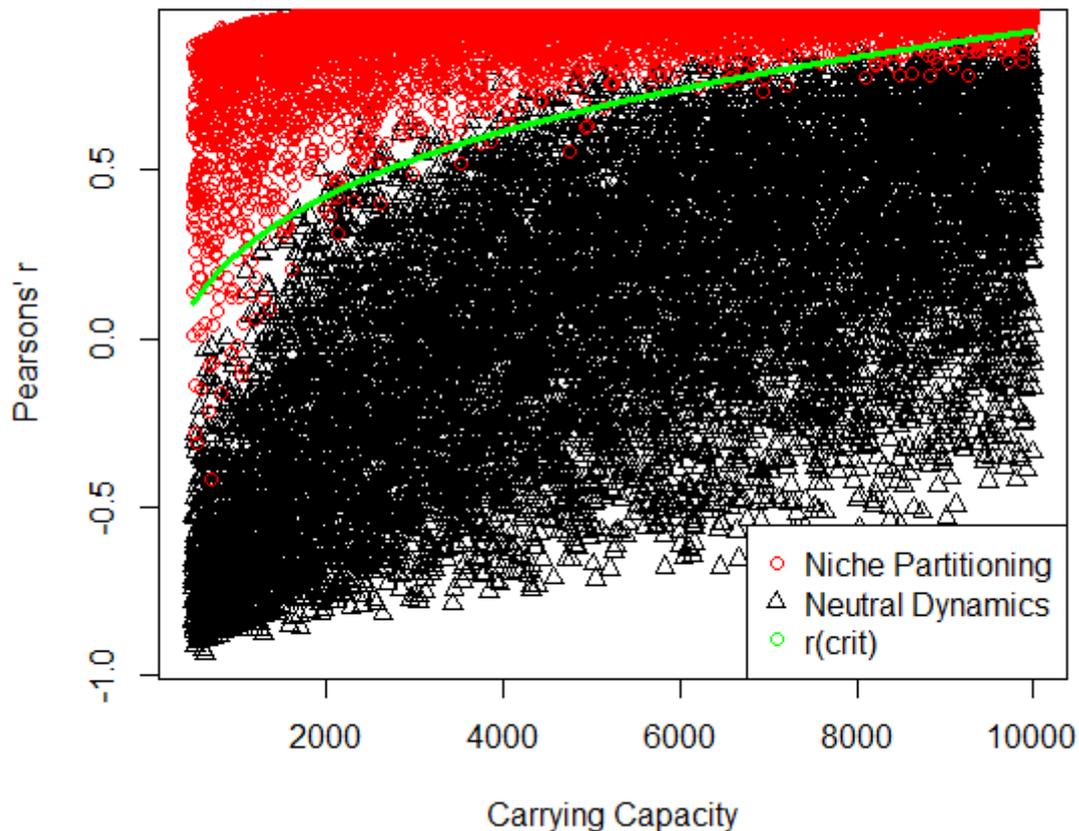


Figure 10: Plot of Pearson's r in dependency to the carrying capacity K . A curve was fitted to the plot in order to separate the values induced by niche partitioning from the ones induced by neutral dynamics. This curve represents the threshold value r_{crit} .

Table 6: Accuracy of the exclusion approach

Alternation	Probability of Success	Confidence Interval (95%)	Most likely Misclassification	Probability of Misclassification
Fixed Parameters	0.940	[0.923; 0.954]	-	-
K	0.847	[0.823; 0.869]	Neutral	
Alpha	0.789	[0.762; 0.814]	NP	0.588
B	0.822	[0.797; 0.845]	RNC/Combination2	0.39 / 0.421
K, Alpha	0.669	[0.639; 0.698]	NP	0.927
K, B	0.657	[0.627; 0.687]	RNC	0.520
Alpha, B	0.685	[0.655; 0.714]	Combination2	0.569
K, Alpha, B	0.525	[0.494; 0.556]	NP	0.944

The accuracy of the identification is given by the probability of success for different parameter alternations. The mechanism which was responsible for the most loss of probability of success is given with “Most likely Misclassification”. The “Probability of Misclassification” gives the value of an exact binomial test with the number of times the mechanism with the highest likeliness for misclassification was not identified and the total number of this mechanism being simulated.

For the cases in which alpha and the carrying capacity were alternated, niche partitioning was scarcely identified. In most of these cases it was misclassified for neutral dynamics. This means the approximation with the root function is not sufficient to keep the correlation values apart.

3.4 Classification via classifier

Mechanisms

The fit created by the binomial classifier was tested to validate its accuracy. To do so I created a matrix of values produced by the three predictors for randomly set mechanisms. For each simulation the mechanism that created these values was written in the matrix as well for later comparison. One thousand simulations were conducted and the classes predicted using the trained fit from before. An exact binomial test yielded a probability of success of 0.967 (Confidence Interval (95%): [0.954, 0.977]). As conclusion it can be said, that a binomial classifier is able to significantly distinguish the four mechanisms

Mechanisms and Combinations

The accuracy of the binomial classifier (this time with 6 predictors) was validated the same way as above. The exact binomial test yielded a probability of success of 0.949 (Confidence Interval (95%): [0.933,0.962]).

Alternating Parameters

As described above, it is necessary for application that the model gives good results with alternating parameters. Table 7 shows how the alternation of parameters influences the accuracy of the identification. If one parameter is alternated it is randomly drawn from its specific interval, as described for the exclusion approach with alternating parameters.

Table 7: Accuracy of the binomial classifier

Alternation	Probability of Success	Confidence Interval (95%)
Fixed Parameters	0.949	[0.933; 0.962]
K	0.970	[0.957; 0.980]
Alpha	0.936	[0.919; 0.958]
B	0.798	[0.772; 0.822]
K, Alpha	0.922	[0.904; 0.938]
K, B	0.825	[0.800; 0.848]
Alpha, B	0.850	[0.826; 0.872]
K, Alpha, B	0.842	[0.812; 0.864]

The model with alternating K-,Alpha- and B- Values was tested a second time with an increased train data set to figure out if it can increase the probability of success. This time, for each mechanism one thousand instead of one hundred simulations provided values for the predictors. The exact binomial test yielded a probability of 0.922 (Confidence Interval (95%): [0.904, 0.938]).

4. Discussion

4.1 Main questions at this study

Is it possible to find characteristic patterns in the time series of population dynamics of different coexistence mechanisms? Is it possible to quantify these patterns in order to distinguish the mechanisms? Is the same true for combinations of these mechanisms? Is the applicability of these results given?

4.2 Answers provided by this study

Each mechanism has a characteristic pattern and it is possible to quantify them. It is also possible to use these patterns for a differentiation between the mechanisms. Not all combinations could be distinguished. For this reason and because the significance of the identification was too low for alternating parameters the applicability is very limited.

4.3 Limitations

Models help us understanding complex processes and allow predictions under certain assumptions. However, because of these assumptions one can never rely one hundred percent on the results given by these models. The same is true here. This study provided a simplified image of processes that happen on a time scale too large for one human life to record. That is why we cannot prove it right or wrong. Instead we need to have a critical look on the presented methods and its flaws.

4.3.1 Parameters that were not alternated

The test for accuracy of the identification was not conducted with an alternation of the parameters $N(1)$, d , and r .

The start value for the population density ($N(1)$) was left out because it is rather trivial to test the behavior of the system with it being alternated. That is because the matter of interest in this study is the long term dynamic of the different mechanisms and not the starting conditions.

The coexistence of the two competing species is very sensible to changes in the values of the parameters d and r . Before calculating the accuracy of the identification, it would have been necessary to check within which interval these parameters can be alternated, so that coexistence is still given for each mechanism and combination. This estimation would have exceeded the time limit of this study.

4.3.2 Sensitivity of RNC towards environmental fluctuations

As mentioned before, it was not possible to create times series for RNC with a standard deviance for the lognormal distribution greater than 0.2, because then coexistence was not given. Relative nonlinearity of competition shows fluctuations with large amplitudes for one species. By increasing the standard deviance the magnitude of environmental impact is increased, leading to an increase in amplitude (not only in RNC but for every mechanism). At some point the amplitude is so strong that the population density equals zero, from which it cannot recover. An increased carrying capacity can postpone that effect but it is still there (Figure 11). In the model used for this study the environment influenced the carrying capacity. Alternatively I briefly examined a similar model in which the environment influenced the intrinsic growth rate r . For this model RNC showed coexistence features for sd – Values up until 0.7. I guess that is because the increase in amplitude for same sd values is smaller in the r -model than in the k -model. That means similar to the increase of the carrying capacity the effect is not eliminated but postponed. As an ecological explanation one might argue that RNC only occurs for small environmental fluctuations because the fluctuation of the overcompensator is

Discussion

rather induced by the dynamics of the species themselves, than exterior factors. In contrast the storage effects' fluctuations are induced by the environment. There are no dynamics between the competing species that would make them fluctuate. That is why the storage effect requires stronger environmental fluctuations. When they are combined (Combination 1) coexistence is given because even when one species' population density is zero, there is still the buffered population growth to keep the reproduction up.

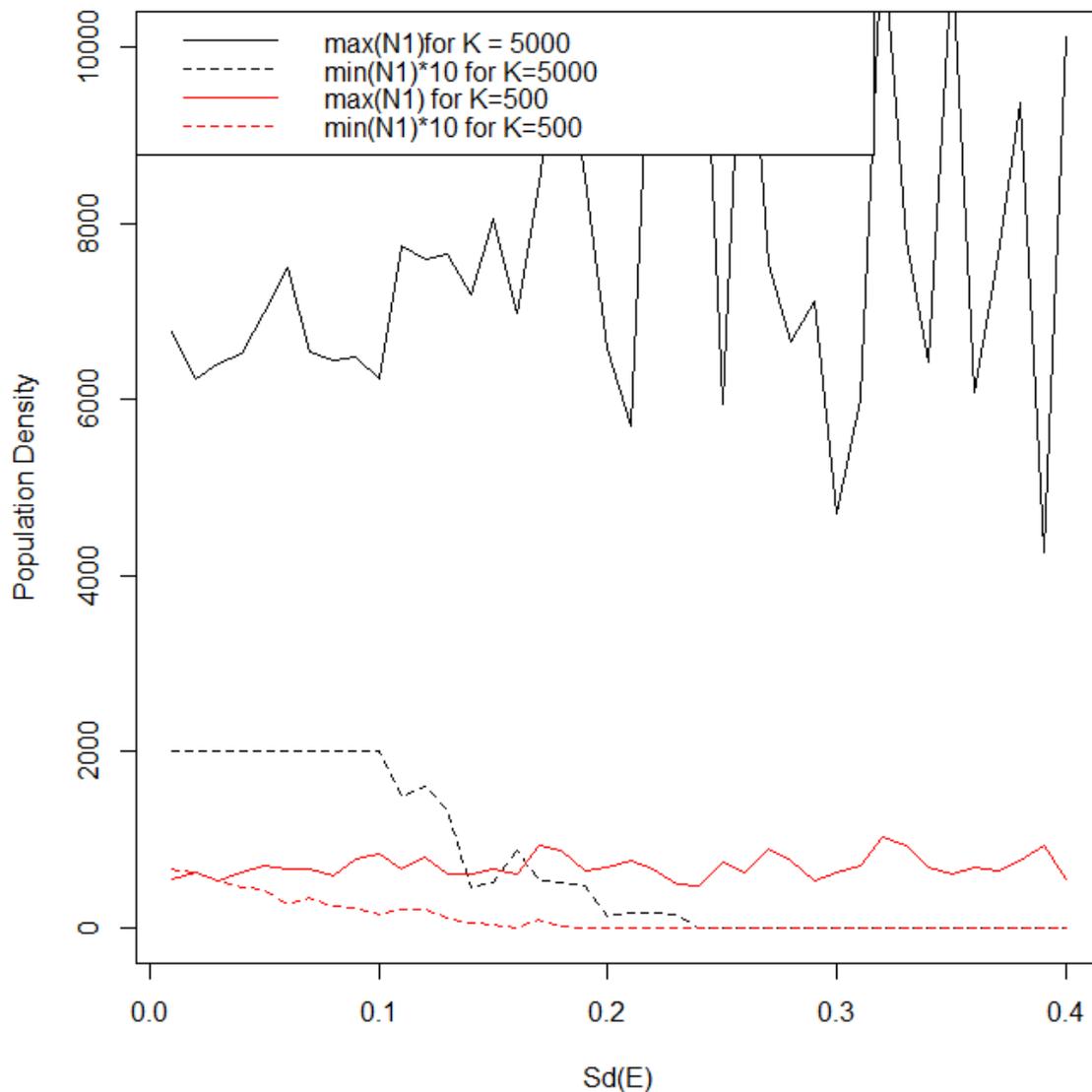


Figure 11: This plot shows the minima and maxima values of population densities for the overcompensating species in the case of RNC, in dependency of the standard deviance value of the lognormal distribution from which the environmental fluctuations are drawn. The minima values are multiplied by ten, so that a tendency is visible on such a large scale. The conclusion of this plot is that with larger carrying capacity (black lines) RNC is able to withstand larger environmental fluctuations without one species going extinct (the dashed black line reaches zero later than the dashed red line).

4.3.3 Distinguishing storage effect and combination 3

As mentioned before, it was not possible to find a tool to differentiate between the storage effect and combination 3. When looking at the population dynamics (Figure 12) it is hard to make out a characteristic pattern. That means the storage effect is the dominant effect here. Because there was no pattern to find, it was not possible to come up with an assessment tool. In the exclusion approach, niche partitioning is identified by calculating the correlation between the competing species. But it is obvious that this relationship is not given anymore because the dynamics are completely different. Without an assessment tool to use as predictor it is very unlikely the binomial classifier is able to tell them apart because the values of the other predictors are too similar (Figure 12). As result of this inability of telling the storage effect and its' combination with niche partitioning apart, the case where all mechanisms are involved is not distinguishable as well.

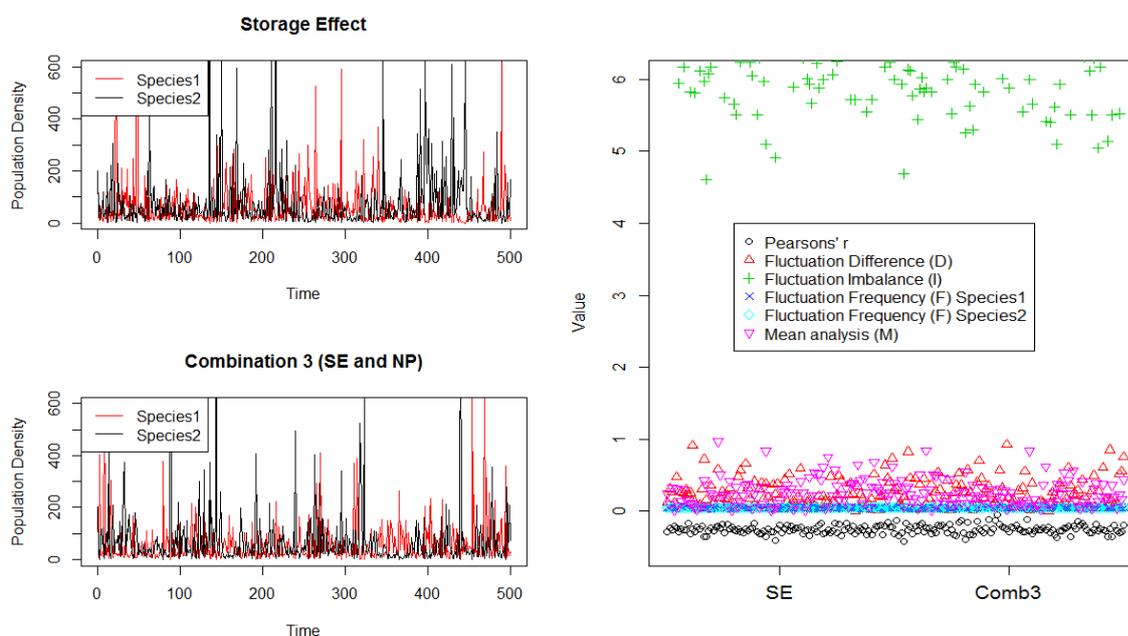


Figure 12: This figure should illustrate the similarity of population dynamics produced by the storage effect and Combination 3. Neither in the patterns of the time series (left), nor in the predictors' values for the binomial classifier (right) a significant difference can be found.

5. Conclusion

As main conclusion of this study it can be said, that it is possible to distinguish coexistence mechanisms by analyzing their times series data, at least to a certain degree. The differentiation of the four mechanisms with fix parameters was significant with both approaches. The identifying of the two combinations that were possible to distinguish yielded good results as well. Still, the failure in identifying Combination 3 leaves the matter unsatisfying unsolved. It would have been a major benefit to be able to identify all the combinations, because presumably combinations of the mechanisms are closer to what happens in nature than single mechanisms. Even though the prediction for fixed parameters is significant, models with fixed parameters are a poor image of nature in reality. Here everything is variable on every possible scale. But as soon as variability is involved, the significance of the prediction goes down (Table 6 and 7). Even so, the results produced by the binomial classifier are better than the ones produced with the exclusion approach. It is interesting that the probability of success for the binomial classifier went remarkably up if the fit was trained with an increased data set. Maybe here is some potential to create a mass of data and apply the trained fit for some field data.

References

- Carroll I.; Nisbet R. (2015) Departures from neutrality induced by niche and relative fitness differences. *Theoretical Ecology*
- Chesson P. (1983) Coexistence in a Stochastic Environment : The Storage Effect. *Population Biology*.
- Chesson P. (1992) Multispecies Competition in Variable Environment. *Theoretical Population Biology* 45, 227-276
- Chesson P. (2003) Quantifying and testing coexistence mechanisms arising from recruitment fluctuations. *Theoretical Population Biology* 64, 345–357
- Chesson P. et.al (2004) Resource pulses, species interactions, and diversity maintenance in arid and semi-arid environments. *Oecologia* 141, 236–253. Springer
- Chesson P. (2000) Mechanisms of maintenance of species diversity. *Annual review of Ecology and Systematics* 31: 343-366.
- Chesson P.; Ellner S. (1989) Invasibility and stochastic boundness in monotonic competition models. *Journal of Mathematical Biology* 27, 117 – 138.
- Elton C. (1927) *Animal ecology*. Sidgwick and Jackson, London, England.
- Mac Arthur (1967) *The theory of the niche*. Population biology and evolution. Pages 159-176
Syracuse University Press, Syracuse, New York, USA.
- Münkemüller T.; Bugmann H.; Johst K. (2009) Hutchinson revisited: Patterns of density regulation and coexistence of strong competitors. *Journal of Theoretical Biology* 259, 109-117
- Grinnell, J. (1917) The niche-relations of the California Thrasher. *Auk* 34:427-433.
- Hartig F.; Munkemüller M.; Johst K.; (2014) On the sympatric Evolution and Evolutionary Stability of Coexistence by Relative Nonlinearity of Competition. *PLoS ONE* 9(9): e94454.
doi:10.1371/journal.pone.0094454
- Holt R. (2001) Species Coexistence. *Encyclopedia of Biodiversity*, Volume 5
- Hutchinson, G. E. (1957) Concluding remarks. *Cold Spring Harbor Symposia . Quantitative Biology* 22:415-427.
- Hutchinson, G. E. (1978) *An introduction to population ecology*. Yale University Press, New Haven, Connecticut, USA.
- Hubbel S. (2001) *The unified Neutral Theory of Biodiversity and Biogeography* . Princeton University Press
- Leibold M.A. (1995) The niche Concept Revisited: Mechanistic Models and Community Context. *Ecology* 76(5), 1371 – 1382
- Sommer U.; Worm B. (2002) *Competition and Coexistence*. ISBN: 978-3-642-62800-9
- Roxburgh S.; Shea K.; Wilson B. (2004) The intermediate disturbance hypothesis: Patch dynamics and mechanisms of species Coexistence. *Ecology* 85(2), 359-371
- Stevens H. (2009) *A Primer of Ecology with R*
- Schöner T.W. (1974) Resource Partitioning in Ecological Communities. *Sciences* vol. 185
- Turelli M. (1981) Niche Overlap and Invasion of Competitors in Random Environments

Appendix

A

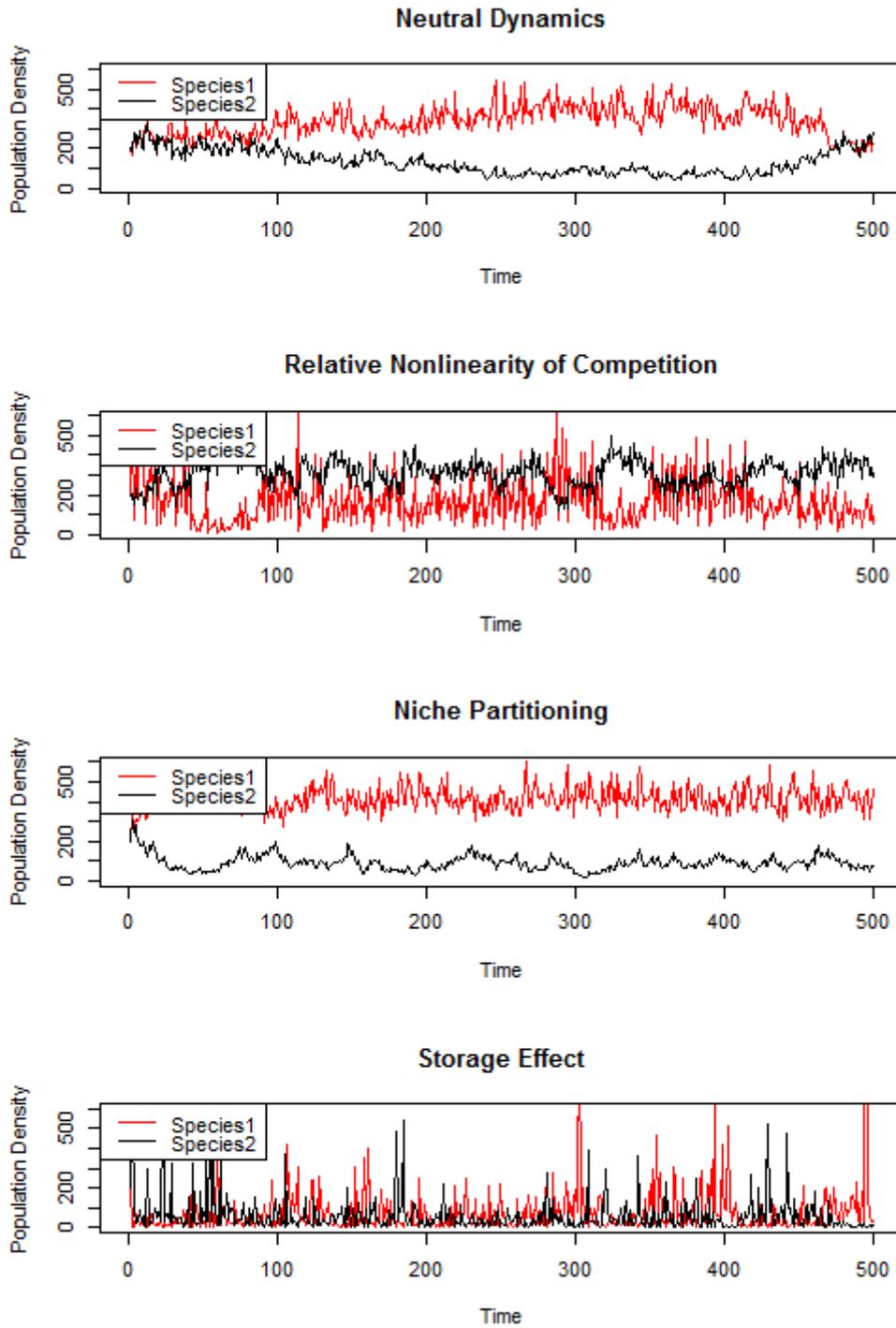


Figure A: Population dynamics for the given Coexistence Mechanisms

B

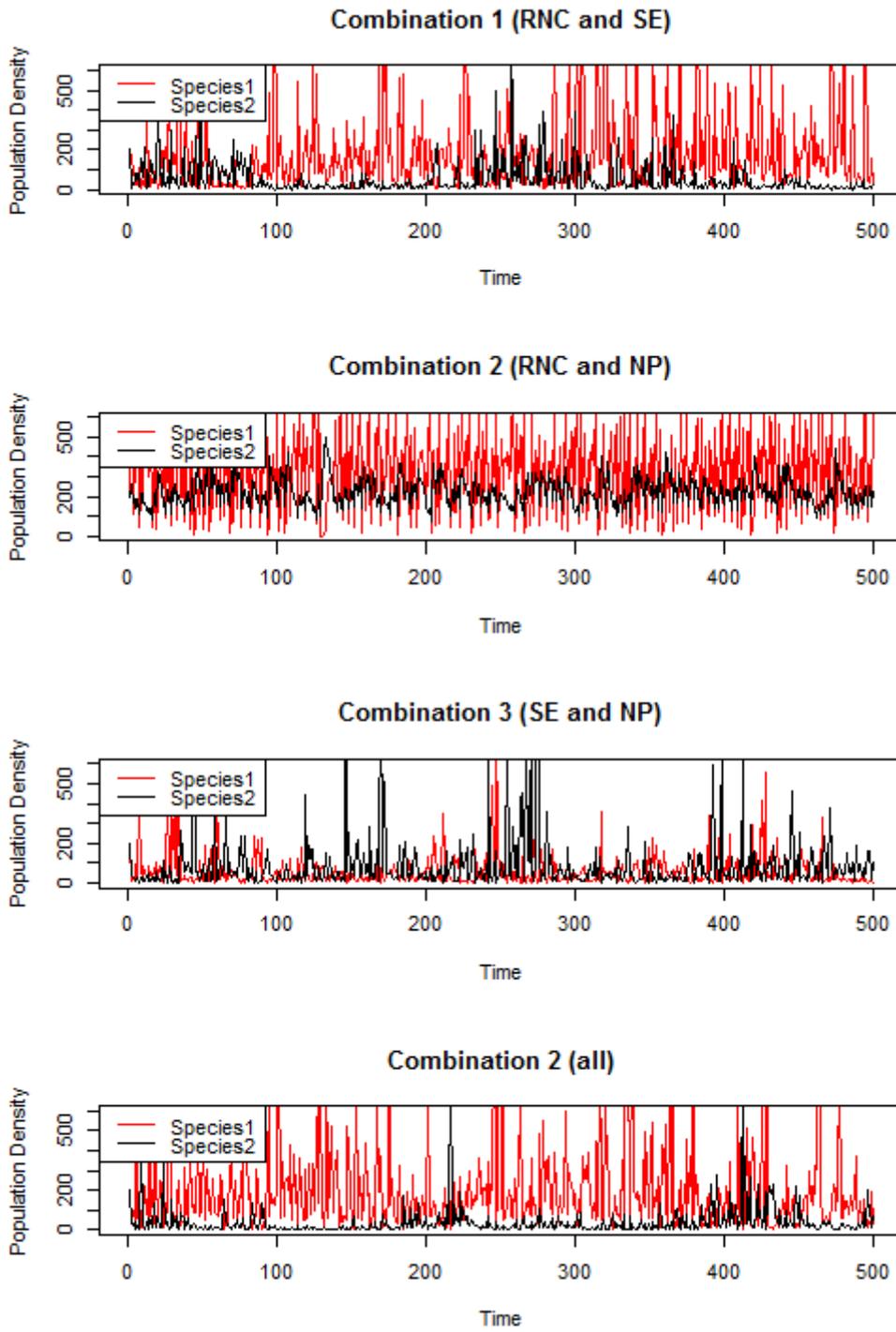


Figure B: Population Dynamics for the given Combinations

R-Code

Mainard Smith and Slatkin with Lotka Volterra and Storage Effect Population Function

This function works as the model described above. It calculates population densities in a time discrete manner. The output consists of the population density, the growth ratio, the difference in population density from one time step to another, the r capita growth rate, and the buffer, for both species. In addition the environment is given as well.

```
msslvspepopfun <- function(parmat, start, time ){

  #Setting required Vectors and Parameters
  N1<- numeric(time)
  N2 <- numeric(time)
  N1[1] <- start[1]
  N2[1] <- start[2]

  fn1 <- numeric(time)
  fn2 <- numeric(time)

  dn1 <- numeric(time)
  dn2 <- numeric(time)

  r.capita1 <- numeric(time)
  r.capita2 <- numeric(time)

  buffer1 <- numeric(time)
  buffer2 <-numeric(time)
  buffer1[1] <- 250000
  buffer2[1] <- 250000

  #Getting Parameter Values from the Parametermatrix
  r1=parmat[1,1]
  d1=parmat[2,1]
  alpha12 = parmat[3,1]
  k1=parmat[4,1]
  b1=parmat[5,1]
  r2=parmat[1,2]
  d2=parmat[2,2]
  alpha21= parmat[3,2]
  k2=parmat[4,2]
  b2=parmat[5,2]

  #Drawing environmental stochasticity
  env <- rlnorm(time,meanlog =0, sdlog =parmat[6,1])

  #Running the for loop for discrete population growth

  for (i in c(1:(time-1))){

    #If the storage Effect is turned on the environmental responses of the competing species are opposing ("different environmental response")
    if(parmat[7,1]==1){env1 <- env[i]
      env2 <- 1/env[i]}
    else{env1 <- env[i]
      env2 <- env[i]}

    a1 <- (r1) / (1 + (r1 -1) * ((N1[i] + alpha12*N2[i])/k1*env1)^b1)
```

Appendix

```

a2 <- (r2) / (1 + (r2 -1) * ((N2[i] + alpha21*N1[i])/k2*env2)^b2)

#Just in case the reproduction ratio gets negative it is set to zero here. This makes sense
because later it is multiplied with N(t) and negative populations do not make sense. In this
case the population is extinct. Also it would create NA's when drawing from the poisson d
istribution.
a1[a1<0]<-0
a2[a2<0]<-0
#If the storage Effect is turned on, buffered population growth is modeled here. If not, a1
and a2 are the reproduction ratios and can just be multiplied with N(t)
if(parmat[7,1]==1)
{

#Filling the buffer (for example producing seeds for a seed bank). The reproduction ratio i
s multiplied by thousand because way more seeds are produced than the actual reproduction r
ate
buffer1[i] <- buffer1[i] + a1*N1[i]*1000
buffer2[i] <- buffer2[i] + a2*N2[i]*1000

#Only a small fraction of the seeds in the seed bank will germinate. Additionally the germi
nation rate is environment dependent.
N1[i+1] <- 0.00005*env1* buffer1[i]
N2[i+1] <- 0.00005*env2* buffer2[i]

#The germinated seeds as well as the destroyed seeds (factor 0.2) are subtracted from the c
urrent buffer
buffer1[i+1] <- buffer1[i] - N1[i+1]
buffer1[i+1] <- buffer1[i] - 0.2*buffer1[i]

buffer2[i+1] <- buffer2[i] - N2[i+1]
buffer2[i+1] <- buffer2[i] - 0.2*buffer2[i]
}
else{
#When storage effect is not involved the adults who died are subtracted here (in the #storag
e effect this has already been taken into account with the factor for destroyed seeds).
N1[i+1] <- N1[i]*(1-d1) *a1
N2[i+1] <- N2[i]*(1-d2) *a2
}

#For demographic stochasticity the resulting population density for the next time step is d
rawn from a poisson distribution

N1[i+1] <- rpois(1,N1[i+1])
N2[i+1] <- rpois(1,N2[i+1])

#storing the reproduction ratio
fn1[i] <- N1[i+1]/N1[i]
fn2[i] <- N2[i+1]/N2[i]
}
#Calculating delta N
dn1 <- N1-c(0,N1[-time])
dn2 <- N2-c(0,N2[-time])
dn1[1] <- NA
dn2[1] <- NA

#Calculating the per capita growth rate
for(k in c(1:time-1)){

r.capita1[k] <- dn1[k+1]/N1[k]
r.capita2[k] <- dn2[k+1]/N2[k]
}

```

Appendix

```
}  
popmat <- cbind(N1,N2,fn1,fn2,dn1,dn2,r.capita1,r.capita2,buffer1,buffer2,env)  
  
return(popmat)  
}
```

The parameters for each mechanism are stored in a parameter matrix which the function can use (parmat). These are the parmats for the different simulations :

```
parmatneutral <- matrix(ncol = 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatneutral[,1] <- c(5,0.05,1,500,1,0.15,0)  
parmatneutral[,2] <- c(5,0.05,1,500,1,0.15,0)  
  
parmatrnc <- matrix(ncol = 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatrnc[,1] <- c(5,0.05,1,500,5,0.15,0)  
parmatrnc[,2] <- c(5,0.05,1,500,1,0.15,0)  
  
parmatlv <- matrix(ncol = 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatlv[,1] <- c(5,0.05,0.6,500,1,0.15,0)  
parmatlv[,2] <- c(5,0.05,0.9,500,1,0.15,0)  
  
parmatse <- matrix(ncol = 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatse[,1] <- c(5,0.05,1,500,1,1.2,1)  
parmatse[,2] <- c(5,0.05,1,500,1,1.2,1)  
  
parmatcomb1 <- matrix(ncol= 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatcomb1[,1] <- c(5,0.05,1,500,5,1.2,1)  
parmatcomb1[,2] <- c(5,0.05,1,500,1,1.2,1)  
  
parmatcomb2 <- matrix(ncol= 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatcomb2[,1] <- c(5,0.05,0.6,500,5,0.15,0)  
parmatcomb2[,2] <- c(5,0.05,0.9,500,1,0.15,0)  
  
parmatcomb3 <- matrix(ncol= 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatcomb3[,1] <- c(5,0.05,0.9,500,1,1.2,1)  
parmatcomb3[,2] <- c(5,0.05,0.6,500,1,1.2,1)  
  
parmatall <- matrix(ncol= 2, nrow = 7)  
#Parametersequence: r,d,alpha,k,b,env,se  
parmatall[,1] <- c(5,0.05,0.6,500,5,1.2,1)  
parmatall[,2] <- c(5,0.05,0.9,500,1,1.2,1)
```

For the exclusion approach it is crucial to estimate threshold values for the assesment tools. This is done by running the model and calculating the assesment tool value for each mechanism 1000 times. For reasons described above Combination 3 and "All" are left out.

1. Imbalance (I)

```
In <- numeric(1000)  
Ir <- numeric(1000)  
Il <- numeric(1000)  
Is <- numeric(1000)  
Ic1 <-numeric(1000)  
Ic2 <-numeric(1000)
```

Appendix

```
for(i in 1:1000){
#Neutral
  pop <- msslvsepopfun(parmat=parmatneutral,start=c(200,200), time = 500)
  In[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
#RNC
  pop <- msslvsepopfun(parmat=parmatrnc,start=c(200,200), time = 500)
  Ir[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
#Niche
  pop <- msslvsepopfun(parmat=parmatlv,start=c(200,200), time = 500)
  Il[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
#Storage Effect
  pop <- msslvsepopfun(parmat=parmatse,start=c(200,200), time = 500)
  Is[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
#Combination 1
  pop <- msslvsepopfun(parmat=parmatcomb1,start=c(200,200), time = 500)
  Ic1[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
#Combination 2
  pop <- msslvsepopfun(parmat=parmatcomb2,start=c(200,200), time = 500)
  Ic2[i] <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))*length
h(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
}
```

2. Fluctuation Difference (D)

```
Dn <- numeric(1000)
Dr <- numeric(1000)
Dl <- numeric(1000)
Ds <- numeric(1000)
Dc1<- numeric(1000)
Dc2<- numeric(1000)

for(i in c(1:1000)){
#Neutral
  pop <- msslvsepopfun(parmat=parmatneutral,start=c(200,200), time = 500)
  pop[pop[,1]==0]<-NA
  pop[pop[,2]==0]<-NA
  Dn[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
#RNC
  pop <- msslvsepopfun(parmat=parmatrnc,start=c(200,200), time = 500)
  pop[pop[,1]==0]<-NA
  pop[pop[,2]==0]<-NA
  Dr[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
#Niche Partitioning
  pop <- msslvsepopfun(parmat=parmatlv,start=c(200,200), time = 500)
  pop[pop[,1]==0]<-NA
  pop[pop[,2]==0]<-NA
  Dl[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
#Storage Effect
  pop <- msslvsepopfun(parmat=parmatse,start=c(200,200), time = 500 )
  pop[pop[,1]==0]<-NA
  pop[pop[,2]==0]<-NA
  Ds[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
#Combination 1
  pop <- msslvsepopfun(parmat=parmatcomb1, start=c(200,200), time = 500 )
  pop[pop[,1]==0]<-NA
  pop[pop[,2]==0]<-NA
  Dc1[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
#Combination 2
  pop <- msslvsepopfun(parmat=parmatcomb2, start=c(200,200), time = 500)
  pop[pop[,1]==0]<-NA
```

Appendix

```
pop[pop[,2]==0]<-NA
Dc2[i] <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))
}
```

3. Correlation

The correlation only needs to be estimated for Neutral dynamics and Niche Partitioning (lv) (see exclusion Schema).

```
rn <- numeric(1000)
r1 <- numeric(1000)

for(i in c(1:1000)){

  neutral <- msslvsepopfun(parmat=parmatneutral,start=c(200,200), time = 500)
  lv <- msslvsepopfun(parmat=parmatlv,start=c(200,200), time = 500)

  rn[i] <- abs(cor(neutral[,1],neutral[,2]))
  r1[i] <- abs(cor(lv[,1],lv[,2]))
}
```

4. Mean Analysis

The mean analysis only needs to be done for the cases Storage Effect and Combination 1 (see exclusion schema)

```
Ms <- numeric(1000)
Mc1 <- numeric(1000)

for(i in c(1:1000)){

  se <- msslvsepopfun(parmat=parmatse,start=c(200,200), time = 500)
  comb1 <- msslvsepopfun(parmat=parmatcomb1, start=c(200,200), time = 500)

  Ms[i] <- abs(1- mean(se[,1])/mean(se[,2]))
  Mc1[i] <- abs(1- mean(comb1[,1])/mean(comb1[,2]))
}
```

5. Oscillation Frequency

Oscillation Frequencies only needs to be calculated for RNC and Combination 2 (see exclusion schema). But because it is not known which one the overcompensator is, it is best to calculate F for both species.

```
Fr <- matrix (nrow = 1000, ncol=2)
Fc2 <- matrix(nrow = 1000, ncol=2)

for(i in 1:1000){

  rnc <- msslvsepopfun(parmat=parmatrnc,start=c(200,200),time = 500)
  comb2 <- msslvsepopfun(parmat=parmatcomb2, start=c(200,200), time = 500)

  Fr[i,1] <-length(which(abs(rnc[,5])<mean(rnc[,1])/2))/length(rnc)
  Fr[i,2] <-length(which(abs(rnc[,6])<mean(rnc[,2])/2))/length(rnc)

  Fc2[i,1] <-length(which(abs(comb2[,5])<mean(comb2[,1])/2))/length(rnc)
  Fc2[i,2] <-length(which(abs(comb2[,6])<mean(comb2[,2])/2))/length(rnc)
}
```

Appendix

Validating Threshold Values for fixed parameters

```
binom.test(length(which(Is>4.5))+ length(which(Ic1 > 4.5)),2000)

##
## Exact binomial test
##
## data: length(which(Is > 4.5)) + length(which(Ic1 > 4.5)) and 2000
## number of successes = 1998, number of trials = 2000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9963924 0.9998789
## sample estimates:
## probability of success
## 0.999

binom.test(length(which(In<4.5))+ length(which(Ir<4.5))+length(which(IL<4.5))+length(which(
Ic2<4.5))),4000)

##
## Exact binomial test
##
## data: length(which(In < 4.5)) + length(which(Ir < 4.5)) + length(which(IL < and 4000
4.5)) + length(which(Ic2 < 4.5)) and 4000
## number of successes = 3985, number of trials = 4000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9938225 0.9978997
## sample estimates:
## probability of success
## 0.99625

binom.test(length(which(Dr>1))+length(which(Dc2>1)), 2000)

##
## Exact binomial test
##
## data: length(which(Dr > 1)) + length(which(Dc2 > 1)) and 2000
## number of successes = 2000, number of trials = 2000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9981573 1.0000000
## sample estimates:
## probability of success
## 1

binom.test(length(which(Dn<1))+length(which(Dl<1))+length(which(Ds<1))+length(which(Dc1<1))
, 4000)

##
## Exact binomial test
##
## data: length(which(Dn < 1)) + length(which(Dl < 1)) + length(which(Ds < and 4000 1)
) + length(which(Dc1 < 1)) and 4000
## number of successes = 3895, number of trials = 4000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9683110 0.9784811
## sample estimates:
```

Appendix

```
## probability of success
##          0.97375

binom.test(length(which(rn>0.5)), 1000)

##
## Exact binomial test
##
## data: length(which(rn > 0.5)) and 1000
## number of successes = 908, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.8883642 0.9251916
## sample estimates:
## probability of success
##          0.908

binom.test(length(which(r1<0.5)), 1000)

##
## Exact binomial test
##
## data: length(which(r1 < 0.5)) and 1000
## number of successes = 999, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.9944411 0.9999747
## sample estimates:
## probability of success
##          0.999

binom.test(length(which(Mc1>0.9)),1000)

##
## Exact binomial test
##
## data: length(which(Mc1 > 0.9)) and 1000
## number of successes = 1000, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.9963179 1.0000000
## sample estimates:
## probability of success
##          1

binom.test(length(which(Ms <0.9)),1000)

##
## Exact binomial test
##
## data: length(which(Ms < 0.9)) and 1000
## number of successes = 979, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.9680777 0.9869548
## sample estimates:
## probability of success
##          0.979

binom.test(length(which(Fc2[,1]<0.03)),1000)

##
## Exact binomial test
##
```

Appendix

```
## data: length(which(Fc2[, 1] < 0.03)) and 1000
## number of successes = 979, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9680777 0.9869548
## sample estimates:
## probability of success
## 0.979

binom.test(length(which(Fr[,1] >0.03)),1000)

##
## Exact binomial test
##
## data: length(which(Fr[, 1] > 0.03)) and 1000
## number of successes = 1000, number of trials = 1000, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.9963179 1.0000000
## sample estimates:
## probability of success
## 1
```

Exclusion Approach Function

In the exclusion function it is randomly decided which mechanism is simulated for every time the function is run. With "TIMES" one can set how many times the model should be simulated. With the letters K, B, A one can decide which parameter should be variable (e.g. B =T means B is drawn randomly)

```
exclusion <- function(TIMES,K=F,B=F,Alpha=F){

#For Later comparison the information which mechanism is drawn gets stored in "Truth". In the vector Guess the result of the analysis is stored.
  Truth <- character(TIMES)
  Guess <- character(TIMES)
  mechanisms <- c("neutral", "rnc", "lv", "se", "comb1", "comb2")
  storage= character(1)

  for(i in c(1:TIMES)){

#First the parameter values are drawn from their specific interval if they are to be alternated.
    if(K==T){capacity <- sample(seq(from=500, to = 1000, by = 100),size = 1)}
    else{capacity <- 500}

    if(B==T){
      b1 <- sample(c(seq(from=0.4, to = 1, by = 0.1),seq(from=3, to = 7, by =0.2)),size = 1)
      if(b1>2.9){ b2 <- sample(seq(from=0.4, to = 1, by =0.1), size =1)}
      else{b2 <- sample(seq(from = 3, to =7, by = 0.2),size=1)}
    }
    else{b1 <- 5
          b2 <- 1}

    if(Alpha==T){alpha1 <- sample(seq(from=0.1, to = 0.9, by =0.02),size=1)
                  alpha2 <- sample(seq(from=0.1, to = 0.9, by =0.02),size=1)}
    else{alpha1 <- 0.6
          alpha2 <- 0.9}

#Then the values are stored in the parmats
```

Appendix

```
parmatneutral <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatneutral[,1] <- c(5,0.05,1,capacity,1,0.15,0)
parmatneutral[,2] <- c(5,0.05,1,capacity,1,0.15,0)

parmatrnc <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatrnc[,1] <- c(5,0.05,1,capacity,b1,0.15,0)
parmatrnc[,2] <- c(5,0.05,1,capacity,b2,0.15,0)

parmatlv <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatlv[,1] <- c(5,0.05,alpha1,capacity,1,0.15,0)
parmatlv[,2] <- c(5,0.05,alpha2,capacity,1,0.15,0)

parmatse <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatse[,1] <- c(5,0.05,1,capacity,1,1.2,1)
parmatse[,2] <- c(5,0.05,1,capacity,1,1.2,1)

parmatcomb1 <- matrix(ncol= 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatcomb1[,1] <- c(5,0.05,1,capacity,b1,1.2,1)
parmatcomb1[,2] <- c(5,0.05,1,capacity,b2,1.2,1)

parmatcomb2 <- matrix(ncol= 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatcomb2[,1] <- c(5,0.05,alpha1,capacity,b1,0.15,0)
parmatcomb2[,2] <- c(5,0.05,alpha2,capacity,b2,0.15,0)

#Here the mechanism for the current simulation is drawn and simulated
Truth[i] <- sample(mechanisms,size=1)
if(Truth[i] == "neutral"){parmat = parmatneutral}
if(Truth[i] == "rnc") {parmat = parmatrnc}
if(Truth[i] == "lv") {parmat = parmatlv}
if(Truth[i] == "se") {parmat = parmatse}
if(Truth[i] == "comb1") {parmat = parmatcomb1}
if(Truth[i] == "comb2") {parmat = parmatcomb2}

pop <- msslvsepopfun(parmat=parmat,start=c(200,200), time = 500)

#The 5 assessment tools are calculated
i1 <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))
i2 <- length(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
I <- i1*i2

r <- cor(pop[,1],pop[,2])
k <- mean(pop[,1])+ mean(pop[,2])
rcrit <- k^(1/5.2877125)/3.0500907 - 0.9590107

f1 <- length(which(abs(pop[,5])<mean(pop[,1])/2))/length(pop)
f2 <- length(which(abs(pop[,6])<mean(pop[,2])/2))/length(pop)

M <- abs(1- mean(pop[,1])/mean(pop[,2]))

#Because in the calculation of the fluctuation difference is a division with the population
density involved, we need to make sure they are not zero (e.g. if one species goes extinct)
. Otherwise the result would be an infinite value.
pop[,1][pop[,1] == 0] <- NA
pop[,2][pop[,2] == 0] <- NA
D <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))

#Now the exclusion of the mechanisms begins(Following the schema).First the mechanisms are
distinguished.
```

Appendix

```
if(I>4.5){Guess[i] <- "se"}
else{
  if(D > 1){ Guess[i] <- "rnc"}
  else{
    if(K!=T){if(abs(r) > 0.5){Guess[i] <- "neutral"}else{Guess[i] <- "lv"}}
    else{if(r < rcrit){Guess[i] <- "neutral"}else{Guess[i] <- "lv"}}
  }
}

#After the involving mechanisms are identified it is possible to determine Combinations.
if(Guess[i] == "se"){if(M > 0.9){Guess[i] <- "comb1"}}
if(Guess[i] == "rnc"){if(f1 < 0.03){Guess[i] <- "comb2"}
  if(f2 < 0.03){Guess[i] <- "comb2"}}
}
result<- cbind(Truth,Guess)
return(result)
}
```

Validating Exclusion Approach

```
x <- exclusion(1000)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.9597851 0.9813153
## sample estimates:
## probability of success
## 0.972

x <- exclusion(1000,K=T)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.8052559 0.8527830
## sample estimates:
## probability of success
## 0.83

x <-exclusion(1000,K=T,Alpha=T)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.8959680 0.9315427
## sample estimates:
## probability of success
## 0.915

x<-exclusion(1000,K=T,B=T)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.6439670 0.7030045
## sample estimates:
## probability of success
## 0.674

x <- exclusion(1000,B=T,Alpha=T)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.6756604 0.7331271
## sample estimates:
## probability of success
## 0.705
```

Appendix

```
x<-exclusion(1000,K=T,B=T,Alpha=T)
binom.test(length(which(x[,1]==x[,2])),1000)

## 95 percent confidence interval:
## 0.7582168 0.8100933
## sample estimates:
## probability of success
## 0.785
```

Classifier

The classifier function is similar to the exclusion function. But instead of the exclusion in the end, this function just yields the values of all the assessment tools in a vector (a). Beside the tools' values (the predictors), for every simulation the mechanism is stored as a "class" in the vector as well. The function "train" is able to determine characteristic patterns in the predictors' values, in dependency to their class. Later this fit can be used on unknown values to assign classes to the characteristic predictor patterns.

```
classifierfun <- function(mechanism, K=F,Alpha=F,B=F){
#First the parameter values are drawn from their specific interval if they are to be altern
ated.
  if(K==T){capacity <- sample(seq(from=500, to = 10000, by = 100),size = 1)}
  else{capacity <- 500}

  if(B==T){
b1 <- sample(c(seq(from=0.4, to = 1, by = 0.1),seq(from=3, to = 7, by =0.2)),size = 1)
if(b1>2.9){ b2 <- sample(seq(from=0.4, to = 1, by =0.1), size =1)}
else{b2 <- sample(seq(from = 3, to =7, by = 0.2),size=1)}
}
else{b1 <- 5
      b2 <- 1}

  if(Alpha==T){alpha1 <- sample(seq(from=0.1, to = 0.9, by =0.02),size=1)
                alpha2 <- sample(seq(from=0.1, to = 0.9, by =0.02),size=1)}
  else{alpha1 <- 0.6
        alpha2 <- 0.9}

#Then the values are stored in the parmats

  parmatneutral <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
  parmatneutral[,1] <- c(5,0.05,1,capacity,1,0.15,0)
  parmatneutral[,2] <- c(5,0.05,1,capacity,1,0.15,0)

  parmatrnc <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
  parmatrnc[,1] <- c(5,0.05,1,capacity,b1,0.15,0)
  parmatrnc[,2] <- c(5,0.05,1,capacity,b2,0.15,0)

  parmatlv <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
  parmatlv[,1] <- c(5,0.05,alpha1,capacity,1,0.15,0)
  parmatlv[,2] <- c(5,0.05,alpha2,capacity,1,0.15,0)

  parmatse <- matrix(ncol = 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
  parmatse[,1] <- c(5,0.05,1,capacity,1,1.2,1)
  parmatse[,2] <- c(5,0.05,1,capacity,1,1.2,1)

  parmatcomb1 <- matrix(ncol= 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
  parmatcomb1[,1] <- c(5,0.05,1,capacity,b1,1.2,1)
  parmatcomb1[,2] <- c(5,0.05,1,capacity,b2,1.2,1)
```

Appendix

```

parmatcomb2 <- matrix(ncol= 2, nrow = 7)
#Parameterreihenfolge: r,d,alpha,k,b,env,se
parmatcomb2[,1] <- c(5,0.05,alpha1,capacity,b1,0.15,0)
parmatcomb2[,2] <- c(5,0.05,alpha2,capacity,b2,0.15,0)

if(mechanism == "neutral") {parmat = parmatneutral}
if(mechanism == "rnc") {parmat = parmatrnc}
if(mechanism == "lv") {parmat = parmatlv}
if(mechanism == "se") {parmat = parmatse}
if(mechanism == "comb1") {parmat = parmatcomb1}
if(mechanism == "comb2") {parmat = parmatcomb2}

#Every mechanism gets its own class
if(mechanism == "neutral") {class <- 1}
if(mechanism == "rnc") {class <- 2}
if(mechanism == "lv") {class <- 3}
if(mechanism == "se") {class <- 4}
if(mechanism == "comb1") {class <- 5}
if(mechanism == "comb2") {class <- 6}

a <- numeric(7)
pop <- msslvsipopfun(parmat=parmat,start=c(200,200), time = 500)

#The 5 assessment tools are calculated
i1 <- length(which(pop[,1]<mean(pop[,1])))/length(which(pop[,1]>mean(pop[,1])))
i2 <- length(which(pop[,2]<mean(pop[,2])))/length(which(pop[,2]>mean(pop[,2])))
I <- i1*i2

r <- cor(pop[,1],pop[,2])

f1 <- length(which(abs(pop[,5]<mean(pop[,1])/2)))/length(pop)
f2 <- length(which(abs(pop[,6]<mean(pop[,2])/2)))/length(pop)

M <- abs(1- mean(pop[,1])/mean(pop[,2]))

#Because in the calculation of the fluctuation difference is a division with the population
density involved, it is crucial to make sure they are not zero (e.g. if one species goes ex
tinct). Otherwise the result would be an infinite value.
pop[,1][pop[,1] == 0] <- NA
pop[,2][pop[,2] == 0] <- NA
D <- abs(mean(abs(pop[,5])/(pop[,1]),na.rm=T) - mean(abs(pop[,6])/pop[,2],na.rm=T))

a[1] <- r
a[2] <- D
a[3] <- I
a[4] <- f1
a[5] <- f2
a[6] <- M
a[7] <- class

return(a)
}

```

The train function needs a train data set and a class set. With the classifier function we stored both in the same vector. But intentionally the class was stored as last. Now we can build up a matrix and use the first columns as train data set and the last as class set. But first, the matrix:

```
classmatrix <- matrix(ncol=7,nrow=600)
```

Appendix

```
for(i in 1:100){
  classmatrix[i,] <- classifierfun(mechanism="neutral")
}
for(i in 1:100){
  classmatrix[100+i,] <- classifierfun(mechanism="rnc")
}
for(i in 1:100){
  classmatrix[200+i,] <- classifierfun(mechanism="lv")
}
for(i in 1:100){
  classmatrix[300+i,] <- classifierfun(mechanism="se")
}
for(i in 1:100){
  classmatrix[400+i,] <- classifierfun(mechanism="comb1")
}
for(i in 1:100){
  classmatrix[500+i,] <- classifierfun(mechanism="comb2")
}
```

The function `Train` comes with the package `"caret"`. The function needs to know which columns are the train data and which are the predictors. Then the fit can be calculated

```
install.packages("caret")
require("caret")
TrainData <- classmatrix[,1:6]
TrainClasses <- classmatrix[,7]

cforestfit <- train(TrainData, TrainClasses,
                    method = "cforest")
```

Now the fit can be checked on a random data set:

```
mat <- matrix(ncol = 7, nrow=1000)
for(i in 1:1000){
  mat[i,]<- classifierfun(mechanism=sample(c("neutral", "rnc", "lv", "se", "comb1", "comb2"),size=1))
}
result <- round(predict(cforestfit, newdata =mat[,1:6]))
check <- cbind(result,mat[,8])
binom.test(length(which(check[,1]==check[,2])),1000)
```

Declaration

I declare to have produced this thesis by myself and to have used no other sources as those listed in the references. Where I have used thoughts from external sources, directly or indirectly, published or unpublished, this is always clearly attributed. All figures were either produced by myself or are attributed with a reference. Furthermore, I certify that this thesis in this or a similar form has not been previously submitted to any other graduation institution

Freiburg, 17.07.2015