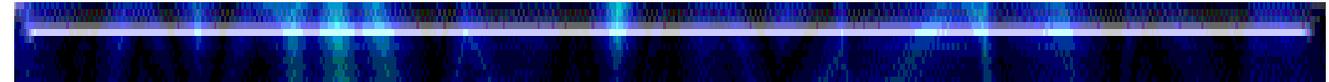


# EcoSimR USER'S MANUAL

## Version 1.00

Nicholas J. Gotelli & Aaron M. Ellison

15 June 2013



## Table of Contents

Introduction.....	3
EcoSimR For R Programmers.....	3
EcoSimR For Beginners.....	3
Setting Up EcoSimR.....	4
Table 1. Files contained in EcoSimR Version 1.00.....	4
A Bestiary of EcoSimR Functions.....	5
Table 2. Functions contained in EcoSimR – General Functions Source.R.....	6
Table 2.1 Function Get.Params.....	7
Table 2.2 Function DataMatrix.....	9
Table 2.3 Function Set.The.Seed.....	10
Table 2.4 Function Null.Model.Engine.....	11
Table 2.5 Function Null.Model.Summary.....	12
Table 2.6 Function Output.Results.....	13
Table 3. Functions contained in EcoSimR – Graphics Source.R.....	14
Table 4.1 Function Null.Model.Plot.....	15
Table 3.2 Function Utilization.Plot.....	16
Table 4. Functions contained in EcoSimR – Algorithms Source.R.....	17
Table 4.1 Function RA1.....	18
Table 4.2 Function RA2.....	19
Table 4.3 Function RA3.....	20
Table 4.4 Function RA4.....	21
Table 5. Functions contained in EcoSimR – Metrics Source.R.....	22
Table 5.1 Function Pianka.....	23
Table 5.2 Function Czekanowski.....	25
Table 5.3 Function Pianka.var.....	27
Table 5.4 Function Czekanowski.var.....	28
Table 5.5 Function Pianka.skew.....	29
Table 5.6 Function Czekanowski.skew.....	31

Literature Cited.....33

## Introduction

Welcome to EcoSimR (“EcoSimmer” as we like to say), a collection of dedicated R functions, algorithms, and metrics for null model analysis. EcoSimR has its roots in EcoSim, a popular program for ecological null model analysis. The free EcoSim program is still available. However, it was coded in the programming language Delphi (a descendant of Pascal), so it can be used only on Windows machines. Although the graphical user interface (GUI) in EcoSim is convenient, GUIs tend to lock the user into a particular set of analyses, and the underlying Delphi code in EcoSim was very difficult to modify. Since the original release of EcoSim before the turn of the century (1998), there have been many exciting new developments in null model analysis; our plan is to update the program and incorporate such changes into EcoSimR. More fundamentally, scientific computing has undergone a “sea-change” with a move towards open-source applications and code, which we heartily embrace. The R platform provides great flexibility and powerful graphics tools for visualizing data and results on Windows, Mac, and Linux operating systems. We will eventually organize EcoSimR into an R package that can be downloaded and installed from the Comprehensive R Archive Network (CRAN) library, but we will always keep a version available at <http://www.uvm.edu/~ngotelli/EcoSim/EcoSim.html>

## EcoSimR For R Programmers

If you are already an experienced R programmer, then you can dig right in and begin accessing EcoSimR's functions for algorithms, metrics, and graphics that you need. The graphics functions in EcoSimR are generic, but they are set up to display the sample data sets, so you will probably need to modify them to properly plot your own data. We recommend that you always copy and rename any of EcoSimR's functions before you start modifying them. If you mess things up, just download a fresh copy of EcoSimR. At this time, the functions in EcoSimR do not require any additional R packages, and all of the graphics routines are based on R's traditional graphics system.

## EcoSimR For Beginners

Even if you have never used R before, you should be able to use EcoSimR for null model analysis. Each major module in EcoSimR includes a “shell program” such as `EcoSimR - Niche Overlap Shell.R`. Running this shell program in R will load a properly formatted sample data set, run an appropriate null model analysis, and generate text and graphic output to your computer screen. The next step is to read through the shell program, where you will find some lines of code that set the options and model parameters. This is where you can easily make changes. For example, you can choose to have the text output displayed on your screen or written to an output file, and you can also create a .pdf file of the graphical output. Changing the null model parameters will give you a new variation on the null model. Run it again and see how the results change, if at all. For each module, we have included a full tutorial that will explain the rationale for the particular null model, and step you through the analysis and interpretation of the output.

If you have not used R before, you will initially miss the convenience of a GUI, with its radio buttons and drop-down menus. Instead, you will have to get used to using an “old-school” Command Line Interface (CLI). With a CLI, you type and submit individual commands in the console at the R prompt, which looks like:

> <type your command here>

or you submit multiple commands with an R script (`filename.R`), which is a simple text file with lines of R code and comments. Once you get the hang of the CLI, you will appreciate the power and flexibility of this style of computing. At that point, you can study the underlying R code in EcoSimR, learn how it works, and eventually write your own functions to do exactly what you want. We sincerely hope that EcoSimR will serve as a gateway drug that will turn you into a full-fledged R addict. To get started down this path, visit the R website to download and install R on your computer: <http://www.r-project.org/>

## Setting Up EcoSimR

**D**ownload the EcoSimR files to your computer and place them all in a single new file folder. If you are using Rstudio (available for free at <http://www.rstudio.com/>) or some other Integrated Development Editor (IDE), you can set up an R project in this folder. If you are using the standard R installation, change the directory (`Change dir...` under the File menu) to this folder so that EcoSimR can find all of the files that it needs. Beginners should load and run `EcoSimR - Niche Overlap Shell.R` (or one of the other `Shell` files) to start interacting with EcoSimR. Experienced programmers should "source" the `EcoSimR - Main Source.R` script, which will load and source the EcoSimR algorithms and metrics into the workspace. Table 1 lists the files that are contained in Version 1.00 of EcoSimR.

**Table 1. Files contained in EcoSimR Version 1.00.**

File Name	Details
<code>EcoSimR - Main Source.R</code>	R script file that sources General Functions, Analytics, Graphics, Algorithms, and Metrics into the R workspace.
<code>EcoSimR - General Functions Source.R</code>	R script file that contains functions for setting the random number generator, summarizing the null model output, converting the input data file into a matrix, and routing the output to the screen or to files.
<code>EcoSimR - Graphics Source.R</code>	R script file that contains functions for graphing the histogram of simulated values and the observed metric from a null model analysis, and shell-specific functions for visualizing the input data matrix and one of the simulated data matrices.
<code>EcoSimR - Algorithms Source.R</code>	R script file that contains a variety of functions for simulating a single null-assemblage from an input matrix. These functions are specific to different null models and have different data structures and arguments.
<code>EcoSimR - Metrics Source.R</code>	R script file that contains a variety of functions for summarizing the pattern in an input matrix, usually as a single number. These functions are specific to different null models and have different data structures and arguments.
<code>MacArthur's</code>	Comma-delimited <code>.csv</code> data file for illustrating niche overlap

Warblers.csv	analysis. Each row is a species of warbler, each column is a different subregion of an idealized coniferous tree, and entries are non-zero real numbers that represent the proportion of time that each species was recorded foraging in a different subregion of a tree. Data from Figures 2-6 in MacArthur (1958).
EcoSimR - Version History.pdf	A .pdf file summarizing the internal updates, major version changes, and release dates for EcoSimR.
Niche Overlap Tutorial.html	A step-by-step guide for running and interpreting null model analyses of niche overlap. Includes a brief discussion of the rationale of the analyses and a brief guide to critical literature citations.

## A Bestiary of EcoSimR Functions

The following tables give a catalog listing of each of the functions contained within EcoSimR's five primary source files: General Functions, Analytics, Graphics, Algorithms, and Metrics. Sub-tables then list the full detail for each EcoSimR function. You should also consult the tutorials for additional information on the use and interpretation of particular algorithms and metrics. For some of the functions, we have used the following color code for function names:

### <FunctionName>

Functions highlighted in green font represent the default choice or valid alternatives for particular EcoSim modules. When all of the default options are used together (as in the Shell scripts), the resulting analysis is well-behaved statistically, and should not be overly prone to Type I errors (rejecting the null hypothesis too frequently for a “random” data set).

### <FunctionName>

Functions highlighted in red font represent algorithms and/or metrics that should be avoided for certain kinds of null model analyses. They usually are prone to Type I errors, and may yield statistical significance for simulated “random” data sets. We have nevertheless included these functions because of their historical significance and because you may want to use them for other kind of randomizations. Moreover, systematically comparing the behavior of a data set with different kinds of randomization algorithms will help you to understand which assumptions are most important in generating patterns.

### <FunctionName>

Functions highlighted in blue font represent algorithms and/or metrics that we think have potential merit, but have not yet been explored in the null models literature. Before publishing with them, you should conduct some of your own benchmark tests with artificial data sets to make sure their behavior is valid (see Gotelli and Ulrich 2012 for additional details).

**Table 2. Functions contained in  
EcoSimR – General Functions Source.R**

Function Name	Purpose
Get.Params	Bundles together a single list of parameters for setting up and running a null model.
Data.Matrix	Reads in a user-supplied .csv file as a matrix.
Set.The.Seed	Sets and stores the random number seed from the system clock or accepts a user-supplied random number seed.
Null.Model.Engine	This function is a generic engine that will run null model simulations. It takes as input a data matrix, a specified number of replicates, a null model algorithm, and a metric. It returns a list of the metric for the data matrix, a vector of metric values for the simulated matrices, a measure of the elapsed time for the simulations, and a unique date stamp for the entire run.
Null.Model.Summary	Generic function that calculates p-values and summary statistics for null model analysis, Time-stamps the analysis, and records the elapsed time.
Output.Results	Takes text and graphical outputs from the null model summary and ports them to the screen and/or to .csv and .pdf files. Optionally displays EcoSimR version information.

**Table 2.1 Function** Get.Params

<b>Function Call</b>	Get.Params (Params)
<b>File Location</b>	EcoSimR - General Functions Source.R
<b>Module</b>	Generic
<b>Arguments</b>	<pre>P1="Macarthur Warblers.csv", P2="Niche Overlap.txt", P3="RA3" P4="Pianka" P5=1000 P6=0 P7="screen" P8="screen" P9="none" P10="Niche.Overlap.Plot"</pre> <p>The Params function bundles up the following list of parameter values that define a null model analysis: P1 is the .csv input file, P2 is the .txt output file, P3 is the algorithm name, P4 is the metric name, P5 is the number of replications, P6 is the random number seed (0 is a seed randomly set by r), P7 is the location for graphic output, P8 is the location for text output, P9 is the location for program information, and P10 is the kind of graphic to be used (which is specific to a particular null model).</p>
<b>Return Values</b>	Params The Params list of variables is passed down to various functions, which use some its elements.
<b>Details</b>	In each shell program, the inputs to Params are set to provide default values for a particular null model analysis. These parameters can be easily changed by the user at the very top of the shell program. Future modules of EcoSimR may add additional elements to this list for functions that may require additional parameters.

**Table 2.2 Function** `DataMatrix`

<b>Function Call</b>	<code>DataMatrix(Data.File)</code>
<b>File Location</b>	EcoSimR - General Functions Source.R
<b>Module</b>	Generic
<b>Arguments</b>	<code>Data.File="Macarthur Warblers.csv"</code> Data.File is a user-supplied character string indicating the name of the input file. The default file name is "Macarthur Warblers.csv"
<b>Return Values</b>	<code>Data.Matrix</code> The function <code>DataMatrix</code> coerces the input <code>.csv</code> data file to a matrix <code>Data.Matrix</code> , with row names taken from the first column and column names (header) taken from the first row.
<b>Details</b>	Most EcoSimR routines use a 2-dimensional numeric matrix as input, and this simple call sets up the matrix. If your data are not in such a form, it is probably simplest to modify them first in a separate script file before calling EcoSimR. There are almost no null model algorithms that account for missing data, so all of the analyses in EcoSimR require that the input data matrix is complete, and will fail (and generate an error message) if the matrix contains "NA" values.

**Table 2.3 Function** Set.The.Seed

<b>Function Call</b>	Set.The.Seed(myseed)
<b>File Location</b>	EcoSimR - General Functions Source.R
<b>Module</b>	Generic
<b>Arguments</b>	<p>Myseed=0</p> <p>an integer value with a default of 0. For the default, a random integer (in the range of <math>\pm 2 \times 10^9</math>) is generated and supplied to the system to set and store the random number seed. Alternatively, the user may specify a random seed; subsequent runs using the same user-supplied seed will return an identical set of results. This is especially useful when learning how the program works.</p>
<b>Return Values</b>	<p>Random.Integer</p> <p>Returns the random integer that was generated to set the random number seed in R, or returns the user-supplied random integer.</p>
<b>Details</b>	<p>This function sets the random number seed by first generating a random integer. This integer is returned as part of the output so that the particular results can be replicated in a future run. Alternatively, this function uses and returns an integer that is supplied by the user.</p>

**Table 2.4 Function** `Null.Model.Engine`

<b>Function Call</b>	<code>Null.Model.Engine(p)</code>
<b>File Location</b>	<code>EcoSimR - Analytics Source.R</code>
<b>Module</b>	Generic
<b>Arguments</b>	<code>p=Param.List</code>  p is a list of input parameters for null model analysis. <code>Null.Model.Engine</code> uses the list elements <code>p\$Data.File</code> (a character string designating the input data file), <code>p\$N.Reps</code> (an integer > 2 indicating the number of replications), <code>p\$Algorithm</code> (a character string designating the randomization algorithm), and <code>p\$Metric</code> (a character designating the metric).
<b>Return Values</b>	<code>Null.Model.Out</code>  <code>Null.Model.Out</code> is a list that contains <code>Obs</code> (the metric value calculated for the input <code>Data.Matrix</code> ), <code>Sim</code> (a vector of metric values calculated for the set of <code>N.reps</code> simulations), <code>Elapsed.Time</code> (a measure of how much time it took to run the entire simulation model), and <code>Time.Stamp</code> (a unique <code>Time.Stamp</code> that is used for all graphic and text file outputs from a single run).
<b>Details</b>	This is a powerful generic function for null model analysis. It takes as input a function call for a randomization algorithm that operates on a matrix, a function call for a metric that is calculated on the matrix, and a constant that defines the number of replicate simulations to run. It runs the null model and returns a list of null model outputs. These can be accessed by the user or are passed to other functions specified by the shell program.

**Table 2.5 Function** `Null.Model.Summary`

<b>Function Call</b>	<code>Null.Model.Summary(Null.Model.Out,Output.File)</code>
<b>File Location</b>	EcoSimR - General Functions Source.R
<b>Module</b>	Generic
<b>Arguments</b>	<p><code>Null.Model.Out=list(Obs=runif(1),Sim=runif(n=1000),Elapsed.Time=NULL,Time.Stamp=NULL),Output.File=NULL</code></p> <p><code>Null.Model.Out</code> is a list that includes the observed metric <code>Obs</code>, a vector of simulated values <code>Sim</code>, elapsed time and time stamp values from the simulation (<code>Elapsed.Time</code>, <code>Time.Stamp</code>). The supplied default values for <code>Null.Model.Out</code> are a single random uniform (0,1) value for <code>Obs</code>, and a vector of 1000 random uniform (0,1) values for <code>Sim</code>. The default value for <code>Elapsed.Time</code>, <code>Time.Stamp</code>, and <code>Output.File</code> is <code>NULL</code>. The default settings print the summary metrics to the console, but output is routed to a file if the user supplies a file name for <code>Output.File</code>.</p>
<b>Return Values</b>	No return values, but summary metrics are calculated and printed, either to the console or to a named output file.
<b>Details</b>	<p>This function sweeps and writes the output from the null model analysis: the time stamp, data file, output file, random number seed, number of replications, elapsed time, the metric, and the algorithm used.</p> <p>It also reports the observed index, the mean and variance of the simulated index, upper and lower 95% confidence intervals (both one- and two-tailed) for the metric, the estimated tail probability for the observed data, and the estimated standardized effect size (SES):</p> $SES = \frac{(X_{obs} - \bar{X}_{sim})}{\sigma_{sim}}$ <p>All of these results are then ported to the screen or to a named output file.</p>

**Table 2.6 Function** `Output.Results`

<b>Function Call</b>	<code>Output.Results(Null.Result, Data.File, Plot.Output, Print.Output, Output.File, Display.About)</code>
<b>File Location</b>	<code>EcoSimR - General Functions Source.R</code>
<b>Module</b>	Generic
<b>Arguments</b>	<code>p=Param.List, Null.Result=Null.Model.Engine()</code>  The variable <code>p</code> contains the <code>Param.List</code> , which has information on whether graphic and text output is to be sent to the screen or to a file. <code>Null.Result</code> is the list of the null model output that is used to create graphical and numerical output for display.
<b>Return Values</b>	No return values
<b>Details</b>	This function ports the summary text output, graphical output, and program information to the screen or to files ( <code>.csv</code> or <code>.pdf</code> ), based on the user's initial specifications in the shell script.

**Table 3. Functions contained in  
EcoSimR – Graphics Source.R**

<b>Function Name</b>	<b>Purpose</b>
<code>Null.Model.Plot</code>	This function is a generic plotting function for a histogram of simulated metric values, an observed metric value, and 1- and 2-tailed 95% confidence intervals.
<code>Utilization.Plot</code>	This function plots the observed utilization matrix and one of the simulated matrices in the niche overlap module. For each matrix, the rows are species, the columns are utilization categories, and the symbols are circles, with the area of the circle proportional to the utilization of a particular resource category by a species.

**Table 4.1 Function** `Null.Model.Plot`

<b>Function Call</b>	<code>Null.Model.Plot(Null.Model.Out, Date.Stamp)</code>
<b>File Location</b>	EcoSimR - Graphics Source.R
<b>Module</b>	Generic
<b>Arguments</b>	<p><code>Null.Model.Out=list(Obs=runif(1), Sim=runif(n=1000)), Date.Stamp=date()</code></p> <p><code>Null.Model.Out</code> is a list that contains <code>Obs</code> (the metric value calculated for the input <code>Data.Matrix</code>), <code>Sim</code> (a vector of metric values calculated for the set of <code>N.reps</code> simulations), and <code>Date.Stamp</code> (a unique <code>Date.Stamp</code> that is used for all graphic and text file outputs from a single run). The default value for <code>Null.Model.Out</code> is a list that contains a single random uniform (0,1) value, and a vector of 1000 random uniform (0,1) values. The default value for <code>Date.Stamp</code> is a call to the current date.</p>
<b>Return Values</b>	None (creates a plot)
<b>Details</b>	This is a generic plotting function for null model analysis. It plots the histogram of metric values calculated for each of the simulated matrices (blue bars), with a vertical red line indicating the metric value for the observed matrix, and dashed vertical black lines to indicate 1- and 2-tailed 95% confidence intervals.

**Table 3.2 Function** Utilization.Plot

<b>Function Call</b>	Utilization.Plot(Data, Algorithm, Date.Stamp, Plot.Output)
<b>File Location</b>	EcoSimR - Analytics Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<pre>Data=matrix(rpois(80,1),nrow=5), Algorithm="RA3", Date.Stamp=date(), Plot.Output="screen"</pre> <p>Data is the utilization matrix that is the input for a niche overlap analysis, Algorithm is the user-selected algorithm for analysis, Date.Stamp is the unique date stamp that is used for all graphic and text file outputs from a single run, and Plot.Output is a vector of character data ("file", "screen", or "none") for porting the graphic output.</p>
<b>Return Values</b>	None (creates a plot)
<b>Details</b>	<p>This graphic routine is specific to the Niche Overlap module. It runs a single simulation of the specified algorithm, plots the observed utilization matrix (in red) and the simulated utilization matrix (in blue) for niche overlap analysis. For each matrix, the rows are species, the columns are utilization categories, and the symbols are circles, with the area of the circle proportional to the utilization of a particular resource category by a species. If the utilization value is 0, no circle is plotted. The default value for Data is a matrix with 5 rows (= species) and 16 columns (= resource states), filled with integers randomly sampled from a poisson distribution (<math>\lambda = 1</math>). The default Algorithm is "RA3", the default Date.Stamp is the current date, and the default Plot.Output is "screen".</p>

**Table 4. Functions contained in  
EcoSimR - Algorithms Source.R**

Function Name	Purpose
RA1	This function randomizes a resource utilization matrix by replacing each entry in the matrix with a value drawn from a random uniform (0,1) distribution.
RA2	This function randomizes a resource utilization matrix by replacing each of the non-zero elements in the matrix with a value drawn from a random uniform (0,1) distribution.
RA3 (Default)	This function randomizes a resource utilization matrix by reshuffling the observed values for each species (= rows) among the different resource categories (= columns)
RA4	This function randomizes a resource utilization matrix by reshuffling the non-zero elements for each species (= rows) among the different resource categories (= columns)

**Table 4.1 Function RA1**

<b>Function Call</b>	RA1 (m)
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><i>m</i> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap metrics. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <i>m</i> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><i>m</i></p> <p><i>m</i> is a randomized resource utilization matrix that has the same dimensions as the original matrix. Each entry in the randomized matrix is drawn from a random uniform (0,1) distribution. Thus, all 0 values in the original matrix are replaced by positive values.</p>
<b>Details</b>	<p>This is the simplest null model algorithm for resource utilization. It assumes that each species in the null assemblage is a “super-generalist” that can potentially use all resource categories in any possible frequency. This “isotropic” null model usually gives highly significant results because real species have limits on their niche breadth and degree of specialization. Moreover, this algorithm is vulnerable to sampling effects. For example, if only a single observation were available for a species, the observed utilization would be 100% specialization on one resource category. Compared to RA1, such a species would appear highly specialized, even if its resource use is random. For this reason, RA1 is “too null” and is prone to Type I errors (falsely rejecting a true null hypothesis). <b>Not recommended.</b></p>

**Table 4.2 Function RA2**

<b>Function Call</b>	RA2 (m)
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<pre>m=matrix(rpois(80,1)nrow=10)</pre> <p>m is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of m is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	m <p>m is a randomized resource utilization matrix that has the same dimensions as the original matrix. Each of the non-zero entries in the randomized matrix is replaced by a random draw from a uniform (0,1) distribution.</p>
<b>Details</b>	<p>This algorithm assumes that as long as a species can utilize a resource, then any possible utilization of that resource is possible. However, if the species never uses the resource in nature, it also does not use it in the null assemblage. Thus, the number and placement of zero values in the original matrix is preserved. This model is more conservative than RA1, although it will still simulate niche breadths that are wider than observed for a species, and it is still vulnerable to sampling effects. <b>Not recommended.</b></p>

**Table 4.3 Function RA3**

<b>Function Call</b>	RA3 (m)
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>m</code></p> <p><code>m</code> is a randomized resource utilization matrix that has the same dimensions as the original matrix. The elements within each row of the matrix are reshuffled equiprobably among the different columns (= resource categories). The elements in each row are reshuffled separately and independently.</p>
<b>Details</b>	<p>This algorithm preserves the observed niche breadth<sub>s</sub> and number of utilized resource categories, but the utilization of each category is random with respect to that of other species. This algorithm provides some control for differences in sampling effort among species by preserving the observed niche breadth of each species. RA3 performed well in benchmark tests by Winemiller and Pianka (1990) and is the default algorithm for this module. <b>Recommended (Default)</b>.</p>

**Table 4.4 Function RA4**

<b>Function Call</b>	RA4 (m)
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<pre>m=matrix(rpois(80,1)nrow=10)</pre> <p>m is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of m is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p>m</p> <p>m is a randomized resource utilization matrix that has the same dimensions as the original matrix. The non-zero elements within each row of the matrix are reshuffled equiprobably among the different columns (= resource categories). The non-zero elements in each row are reshuffled separately and independently.</p>
<b>Details</b>	<p>Like RA3, this algorithm preserves the observed niche breadth and number of utilized resource categories, but with the added restriction that the pattern of 0s in the original matrix, which represent un-utilized resource categories, is preserved in the simulated matrix. Because RA4 is the most constrained of the four niche overlap algorithms, it will generate null matrices that are more similar in structure to the original data. This is a good safeguard against false positives (Type I error), but it may entail a loss of statistical power (Type II error). <b>Recommended.</b></p>

**Table 5. Functions contained in  
EcoSimR – Metrics Source.R**

Function Name	Purpose
<code>Pianka (Default)</code>	This function calculates the Pianka niche overlap index for all unique pairs of species in a utilization matrix. It returns the average overlap among all pairs.
<code>Czekanowski</code>	This function calculates the Czekanowski niche overlap index for all unique pairs of species in a utilization matrix. It returns the average overlap among all pairs.
<code>Pianka.var</code>	This function calculates the Pianka niche overlap index for all unique pairs of species in a utilization matrix. It returns the variance of niche overlap overlap among all pairs.
<code>Czekanowski.var</code>	This function calculates the Czekanowski niche overlap index for all unique pairs of species in a utilization matrix. It returns the variance of niche overlap overlap among all pairs.
<code>Pianka.skew</code>	This function calculates the Pianka niche overlap index for all unique pairs of species in a utilization matrix. It returns the skewness of niche overlap among all pairs.
<code>Czekanowski.skew</code>	This function calculates the Czekanowski niche overlap index for all unique pairs of species in a utilization matrix. It returns the skewness of niche overlap among all pairs.

**Table 5.1 Function `Pianka`**

<b>Function Call</b>	<code>Pianka(m)</code>
<b>File Location</b>	<code>EcoSimR - Algorithms Source.R</code>
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1),nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>Pianka</code></p> <p>The Pianka index is a single numeric value on a 0 to 1 scale that summarizes the average pairwise niche overlap in an assemblage. For species 1 with <math>i = 1</math> to <math>n</math> resource categories, the proportional resource utilization <math>p_{1i}</math> of resource state <math>i</math> by species 1 is defined such that:</p> $\sum_{i=1}^n p_{1i} = 1.0$ <p>Niche overlap indices are always calculated by first dividing each entry in the utilization matrix by the corresponding row total, so that all entries are proportional utilization values (<math>p</math>) for each species.</p> <p>The Pianka niche overlap index <math>O</math> for the pair of species (1,2) is calculated as</p> $O_{12} = O_{21} = \frac{\sum_{i=1}^n p_{2i} p_{1i}}{\sqrt{\sum_{i=1}^n p_{2i} \sum_{i=1}^n p_{1i}}}$ <p>The <code>Pianka</code> function returns the average niche overlap calculated over all unique species pairs in the matrix.</p>

**Details**

The Pianka Index is a symmetrical overlap index for a pair of species that quantifies niche overlap among a set of discrete utilization categories and ranges between 0.0 (no overlap) and 1.0 (complete overlap). This index was proposed by Pianka (1973), and is a modified version of MacArthur and Levins' (1967) index, which was originally related to the species interaction coefficients  $\alpha$  from the Lotka-Volterra competition equations. Although that relationship is questionable (Lawlor 1980), the index has sensible numerical properties. It is highly correlated with the Czekanowski index, and either one will give similar results as a simple measure of pairwise niche overlap. **Recommended (Default).**

**Table 5.2 Function** Czekanowski

<b>Function Call</b>	Czekanowski (m)
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p>Czekanowski</p> <p>The Czekanowski index is a single numeric value on a 0 to 1 scale that summarizes the average pairwise niche overlap in an assemblage. For species 1 with <math>i = 1</math> to <math>n</math> resource categories, the proportional resource utilization <math>p_{1i}</math> of resource state <math>i</math> by species 1 is defined such that:</p> $\sum_{i=1}^n p_{1i} = 1.0$ <p>Niche overlap indices are always calculated by first dividing each entry in the utilization matrix by the corresponding row total, so that all entries are proportional utilization values (<math>p</math>) for each species.</p> <p>The Czekanowski niche overlap index <math>O</math> for the pair of species (1,2) is calculated as</p> $O_{12} = O_{21} = 1 - 0.5 \sum_{i=1}^n  p_{2i} - p_{1i} $ <p>The <code>Czekanowski</code> function returns the average niche overlap calculated over all unique species pairs in the matrix.</p>
<b>Details</b>	The Czekanowski Index is a symmetrical overlap index for a pair of species that quantifies niche overlap among a set of discrete utilization

	categories and ranges between 0.0 (no overlap) and 1.0 (complete overlap). This index has frequently been proposed as a measure of niche overlap (references in Hurlbert 1978), and it represents the intersection of the areas of overlap of two utilization functions. When used with the same randomization algorithm, it gives very similar results to the Pianka index. <b>Recommended.</b>
--	---

**Table 5.3 Function `Pianka.var`**

<b>Function Call</b>	<code>Pianka.var(m)</code>
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>Pianka.var</code></p> <p>The Pianka variance index is a non-negative real number that measures the sample variance among species pairs in the Pianka niche overlap index. See the <code>Pianka</code> function for details on the calculation of the underlying index.</p>
<b>Details</b>	<p>The sample variance in niche overlap is a measure of how much heterogeneity there is among individual pairs of species in their overlap. If the sample variance is significantly larger than expected by chance, then the utilization matrix consists of some species pairs with very high overlap and some species pairs with very low overlap (Winemiller and Pianka 1990). In such cases, it may be instructive to examine overlap values for individual pairs of species. <a href="#">Worth exploring.</a></p>

**Table 5.4 Function `Czekanowski.var`**

<b>Function Call</b>	<code>Czekanowski.var(m)</code>
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>Czekanowski.var</code></p> <p>The Czekanowski variance index is a non-negative real number that measures the sample variance among species pairs in the Czekanowski niche overlap index. See the <code>Czekanowski</code> function for details on the calculation of the underlying index.</p>
<b>Details</b>	<p>The sample variance in niche overlap is a measure of how much heterogeneity there is among individual pairs of species in their overlap. If the sample variance is significantly larger than expected by chance, then the utilization matrix consists of some species pairs with very high overlap and some species pairs with very low overlap (Winemiller and Pianka 1990). In such cases, it may be instructive to examine overlap values for individual pairs of species. <a href="#">Worth exploring.</a></p>

**Table 5.5 Function `Pianka.skew`**

<b>Function Call</b>	<code>Pianka.skew(m)</code>
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>Pianka.skew</code></p> <p>The <code>Pianka.skew</code> index is a real number (positive or negative) that measures the skewness among species pairs in the <code>Pianka</code> niche overlap index. See the <code>Pianka</code> function for details on the calculation of the underlying index.</p>
<b>Details</b>	<p>The sample skewness in niche overlap is a measure of how much asymmetry there is among individual pairs of species in their overlap. If the sample skewness is significantly larger than expected by chance, then distribution of pairwise overlap values is right-skewed, with a relatively long right-hand tail of species pairs with high niche overlap. Alternatively, if the sample skewness is significantly smaller than expected by chance, then the distribution of pairwise overlap values is left-skewed, with a relatively long left-hand tail of species pairs with low niche overlap.</p> <p>The sample skewness of niche overlap <math>O</math> is calculated for a set of <math>k</math> species pairs as:</p> $Skewness_o = \frac{\sum_{i=1}^k (O - \bar{O})^3}{\left(\sum_{i=1}^k (O - \bar{O})^2 / (k - 1)\right)^{3/2}}$ <p>Although the individual niche overlap values are bounded between 0.0</p>

	<p>and 1.0, calculated skewness may be extreme compared to the distribution generated by the randomization algorithm. The variance, skewness, and other moments of the distribution are relatively unstudied, but they represent potential ways that pairwise niche overlap values may deviate from random expectations. <a href="#">Worth exploring.</a></p>
--	---



**Table 5.6 Function `Czekanowski.skew`**

<b>Function Call</b>	<code>Czekanowski.skew(m)</code>
<b>File Location</b>	EcoSimR - Algorithms Source.R
<b>Module</b>	Niche Overlap
<b>Arguments</b>	<p><code>m=matrix(rpois(80,1)nrow=10)</code></p> <p><code>m</code> is a resource utilization matrix. Each row is a species, each column is a resource category (unordered), and the entries are the utilization of each resource category by a species. Row and column names may be present from the original input file, but are not used in the analysis. The entries are non-negative real numbers that may reflect counts, percentages, biomass, time, or other units that represent resource utilization. It is not necessary that the row sums add up to 100%, although they will be standardized in this way by the analysis for computation of overlap. Zero entries are allowed, and they indicate that a particular resource category was never used by a species. The default value of <code>m</code> is a matrix of integers with 10 rows and 8 columns. Each entry is a random draw from a Poisson distribution (<math>\lambda = 1</math>).</p>
<b>Return Values</b>	<p><code>Czekanowski.skew</code></p> <p>The Czekanowski skew index is a real number (positive or negative) that measures the skewness among species pairs in the Czekanowski niche overlap index. See the <code>Czekanowski</code> function for details on the calculation of the underlying index.</p>
<b>Details</b>	<p>The sample skewness in niche overlap is a measure of how much asymmetry there is among individual pairs of species in their overlap. If the sample skewness is significantly larger than expected by chance, then distribution of pairwise overlap values is right-skewed, with a relatively long right-hand tail of species pairs with high niche overlap. Alternatively, if the sample skewness is significantly smaller than expected by chance, then the distribution of pairwise overlap values is left-skewed, with a relatively long left-hand tail of species pairs with low niche overlap.</p> <p>The sample skewness of niche overlap <math>O</math> is calculated for a set of <math>k</math> species pairs as:</p> $Skewness_O = \frac{\sum_{i=1}^k (O_i - \bar{O})^3}{\left(\sum_{i=1}^k (O_i - \bar{O})^2 / (k-1)\right)^{3/2}}$ <p>Although the individual niche overlap values are bounded between 0.0</p>

and 1.0, calculated skewness may be extreme compared to the distribution generated by the randomization algorithm. The variance, skewness, and other moments of the distribution are relatively unstudied, but they represent potential ways that pairwise niche overlap values may deviate from random expectations. [Worth exploring.](#)

## Literature Cited

Gotelli, N.J. and W. Ulrich. 2012. Statistical challenges in null model analysis. *Oikos* 121: 171-180.

Hurlbert, S.H. 1978. The measurement of niche overlap and some relatives. *Ecology* 59: 67-77.

Lawlor, L.R. 1980. Overlap, similarity, and competition coefficients. *Ecology* 61: 245-251.

MacArthur, R.H. 1958. Population ecology of some warblers of northeastern coniferous forests. *Ecology* 39: 599-699.

MacArthur, R.H. And R. Levins. 1967. The limiting similarity, convergence, and divergence of coexisting species. *American Naturalist* 101: 377-385.

Pianka, E.R. 1973. The structure of lizard communities. *Annual Review of Ecology and Systematics* 4: 53-74.

Winemiller, K.O. and E.R. Pianka. 1990. Organization in natural assemblages of desert lizards and tropical fishes. *Ecological Monographs* 60: 27-55.