

Short communication

Statistical modelling of communities and ecosystems using the LAMBDA software tool

Steven V. Viscido^{a,*}, Elizabeth E. Holmes^b^a Department of Life Sciences, Winston-Salem State University, 601 S. Martin Luther King Jr. Drive, Winston-Salem, NC 27110, USA^b Conservation Biology Division, Northwest Fisheries Science Center, 2725 Montlake Blvd E., Seattle, WA 98112, USA

ARTICLE INFO

Article history:

Received 3 November 2009

Received in revised form

24 May 2010

Accepted 25 May 2010

Available online 19 June 2010

Keywords:

Multivariate auto-regressive first-order model

Community model

Time series

Interaction matrix

Simulated annealing

ABSTRACT

Understanding species interactions is critical to discovering community dynamics. Recently, statistical methods for estimating species interaction strengths from time series data have been developed based on multivariate auto-regressive first-order, or MAR(1), models. However, the complex coding required presents a substantial barrier for most ecologists. We have developed LAMBDA, a software program that allows users to easily fit MAR(1) models to multi-species time series data. The LAMBDA package covers: data input and transformation, selection of the interactions to include via a search algorithm and model selection, estimation of interaction parameters via conditional least squares (CLS) regression or two different maximum-likelihood (ML) algorithms, estimation of confidence intervals via bootstrapping, and computation of community stability properties using the estimated model. We describe performance tests on the variability of estimates, computation speed, and CLS versus ML estimation using simulated data.

© 2010 Elsevier Ltd. All rights reserved.

Software availability

Name of software: LAMBDA

Developer: Steven V. Viscido

Contact information: Department of Life Sciences, Winston-Salem State University, Winston-Salem, NC 27110, USA. Tel.: +336 750 2216. viscidost@wssu.edu

Year first available: 2007

Availability and cost: Open source; <http://lambda-toolkit.sourceforge.net>

Program language: MatLab version 7 or later

Program size: 2.5 MB

1. Introduction

Ives et al. (2003) developed a statistical framework for the analysis of ecological community dynamics using time series abundance data. This framework uses multivariate auto-regressive first-order, or MAR(1), models, to estimate the strength of interactions among species and between species and environmental covariates. The framework also allows ecologists to estimate a variety of community stability properties. Although this approach

provides a powerful framework for studying community dynamics, implementation requires an intensive programming effort.

We created a MatLab toolbox, LAMBDA, which allows users to perform a full MAR(1) statistical analysis through a graphical user interface (GUI). LAMBDA includes all aspects of the data analysis including data input, parameter estimation, bootstrap confidence intervals, model selection, and diagnostic tools described by Ives et al. (2003). LAMBDA provides ecologists with a statistical software package that allows easy implementation of modern analysis tools for studying and forecasting ecological communities and ecosystems. In addition, LAMBDA is modular and open source, so that as improved algorithms are developed, they can easily be added.

2. Theoretical background

The MAR(1) model is built on a stochastic, univariate, first-order auto-regressive, or AR(1), model for each species:

$$X_t = a + bX_{t-1} + \mathbf{C}\mathbf{U}_{t-1} + E_t \quad (1)$$

This is a density-dependent population model known as the Gompertz model. X_t is the natural log of species abundance at time t , a is the growth rate at 0 density, b is density dependence, \mathbf{U}_t is a $q \times 1$ vector of environmental covariates, and \mathbf{C} is a $1 \times q$ vector of covariate interaction terms. E_t is the process error representing the temporally random deviations in population growth.

* Corresponding author. Tel.: +1 336 750 2216.

E-mail address: viscidost@wssu.edu (S.V. Viscido).

To model a community of p interacting species, we convert Eq. (1) to a multivariate, MAR(1), model (Ives et al., 2003):

$$\mathbf{X}_t = \mathbf{A} + \mathbf{B}\mathbf{X}_{t-1} + \mathbf{C}\mathbf{U}_{t-1} + \mathbf{E}_t \quad (2)$$

\mathbf{X}_t is a $p \times 1$ vector of the log-abundances of each species at time t , \mathbf{A} is a $p \times 1$ vector of species growth rates, \mathbf{B} is the $p \times p$ interaction matrix where b_{ij} is the effect of species j on species i , and \mathbf{C} is the $p \times q$ covariate interaction matrix. The process errors, \mathbf{E}_t , are modeled as multivariate normal with mean 0 and variance matrix Σ . The MAR(1) model as with any regression model, can be used in two ways: (1) to estimate community interaction parameters from an existing dataset, and (2) to forecast community dynamics by simulation, once those parameters are known.

A MAR(1) model also allows ecologists to compute the multivariate stationary distribution of the species abundances. The distribution has a mean μ_∞ and variance-covariance matrix \mathbf{V}_∞ which can be computed from the model parameters. From μ_∞ and \mathbf{V}_∞ , a variety of community stability properties can then be calculated: (a) the variance of the stationary distribution relative to the variance of the process error Σ ; (b) the asymptotic return rate to μ_∞ after a perturbation; (c) the asymptotic return rate of the variance to \mathbf{V}_∞ after a perturbation; and (d) the reactivity, which measures the sensitivity of the community to perturbations. For a full treatment of MAR(1) theory including community stability properties refer to Ives et al. (2003).

3. The LAMBDA toolbox

LAMBDA is a GUI and computational backend written in the MatLab programming language (Mathworks, 2005) that performs all the calculations needed to estimate a MAR(1) model from multivariate time series data, compute statistics, and perform diagnostics. The main output from LAMBDA is an estimated MAR(1) model along with the stability properties (a) through (d), above. The 95% bootstrapped confidence intervals on both the parameters and stability metrics are also given, along with the set of interactions (cf. Ives et al., 2003 and LAMBDA documentation). Although space limitations prohibit a detailed description of the LAMBDA work environment, the software has an extensive manual which includes tutorials (<http://lambda-toolkit.sourceforge.net>).

LAMBDA consists of four interacting elements: (1) data routines for importing, editing, pooling and transforming time series data, (2) estimation of the MAR(1) parameters, (3) model selection search routines for selecting the best-fit constrained interaction matrices, and (4) parametric bootstrapping routines to provide confidence intervals for estimated parameters and stability metrics.

3.1. Data input and manipulation

As input, LAMBDA requires an ASCII file of columns representing the time series of abundance estimates for each species and for any covariates. In general, because MAR(1) models are on the log-scale, transforming the data by the natural logarithm is necessary. To avoid log-of-zero errors, LAMBDA can perform $\ln(x+1)$, $\ln(x+\min(x))$ or $\ln(x+\min(x)/2)$ transformations as specified by the user. LAMBDA can also perform other transformations, including converting the data to z-scores (Hampton and Schindler, 2006).

3.2. Parameter estimation

LAMBDA allows the user to select classical conditional least squares (CLS) regression or maximum-likelihood (ML) estimation for parameter estimation. These methods assume non-

autocorrelation of errors, and additionally the ML techniques assume normality of errors.

3.2.1. Conditional least squares estimation

The conditional least squares (CLS) estimates of MAR(1) parameters are those values of \mathbf{A} , \mathbf{B} , and \mathbf{C} that minimize the squared difference between the observed values \mathbf{X}_t , and those predicted at time t by the MAR(1) model, conditional on the previous values, \mathbf{X}_{t-1} . The CLS algorithm is given in Ives et al. (2003).

3.2.2. Maximum-likelihood estimation

The alternative to CLS estimation is maximum-likelihood (ML) estimation. ML parameters are those parameters that maximize the likelihood of data conditioned on the model. The Nelder–Mead simplex algorithm, which was used in Ives et al. (2003), is provided as an option in LAMBDA. To compute the maximum-likelihood parameters using the simplex method, LAMBDA performs an unconstrained nonlinear optimization, beginning with random \mathbf{A} , \mathbf{B} , and \mathbf{C} matrices, and iterates repeatedly until the minimum value of the negative log-likelihood is reached. Unfortunately, this method takes much longer than CLS estimates, and can become “trapped” in a local minimum.

To cope with these problems, LAMBDA includes an alternative ML method, simulated annealing. Simulated annealing algorithms are based on dynamically jumping around the parameter space looking for the minimum (Cerny, 1985; Kirkpatrick et al., 1983) and slowly reducing jump size until the search arrives at the lowest negative log-likelihood. The number of iterative parameter searches K can be set by the user, and LAMBDA defaults to $K = 10^4$. We let τ represent the “temperature” of the system. Initially $\tau = 10$, and it is then decremented by 1% every $K/100$ iterations.

LAMBDA’s annealing algorithm begins with a random proposed parameter set $P = \{\mathbf{A}, \mathbf{B}, \mathbf{C}, \Sigma\}$ which has a negative log-likelihood, L . Then a proposed parameter draw P' is created where each parameter element in P is multiplied by an independent random -10% to $+10\%$ adjustment drawn from a uniform distribution. The negative log-likelihood L' of the model with P' is then computed. The proposed parameter set is accepted with probability β , where

$$\beta = e^{[(L-L')/\tau]} \quad (3)$$

If the proposed parameter set is accepted, then P' replaces P , and L' replaces L , and the process is repeated for the next parameter, and iterated K times. τ is lowered every $K/100$ iterations and the search converges gradually on the lowest L . The annealing method robustly finds the ML values and takes only slightly longer than CLS estimates for the same number of parameters.

3.3. Constraining the interaction matrices

If we allow every element within the interaction matrices, \mathbf{B} and \mathbf{C} , to be non-zero, then we are presuming that all species and covariates in the system interact. In a natural community, many species will interact so weakly that their interaction strength is effectively zero. From a statistical perspective, elements are “effectively zero” if their presence does not improve the fit of the model. If we can reduce the number of parameters by forcing certain interactions to be zero *without reducing the fit of the model*, then we can improve the estimate quality for the non-zero interactions. LAMBDA provides two ways to do this. First, the investigator can manually define which interactions to include and which to exclude. Second, LAMBDA can perform an iterative search, zeroing out different \mathbf{B} and \mathbf{C} elements to find the model that most parsimoniously explains the data (Ives et al., 2003;

Hampton and Schindler, 2006). The algorithm creates a large number (default 500) of models in which different **B** and **C** elements are randomly set to zero. Each model is fit to the data and the model with either the lowest Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), at the user's discretion, is chosen.

3.4. Bootstrapped confidence intervals

LAMBDA incorporates parametric bootstrapping (Ives et al., 2003) to calculate confidence intervals on estimated parameters. Parametric bootstrapping uses the estimated model to simulate new bootstrapped datasets. The parameters are estimated from each bootstrapped dataset to create a set of bootstrapped parameter sets.

4. LAMBDA performance tests

We examined the variability of the parameter estimates and stability properties using simulated data. The data were simulated using MAR(1) models for two different lake communities: Peter Lake, WI (Ives et al., 2003) and Lake Washington, WA (Hampton and Schindler, 2006). The Peter Lake dataset (Fig. 1) was used to examine the bias and variability of the parameter estimates. The Lake Washington dataset was used to determine the speed and scaling of the three estimation methods versus the community size p .

The Peter Lake community has four interacting species (Ives et al., 2003; Table 1A). Using LAMBDA's simulation module, we simulated 100 replicate time series for the Peter Lake plankton community. For the simulation, we used the **A** estimates ($\mathbf{A} = [0.6497, 1.0628, 1.5055, 2.4089]$) and **B** estimates (Table 1A) for Peter Lake but did not include any covariates ($\mathbf{C} = 0$) in this simulation. We used simulations with process error variance $\Sigma = \sigma^2 \mathbf{I}$, where \mathbf{I} is the $p \times p$ identity matrix and σ^2 was drawn from a random uniform distribution with a mean 0 and variance of 1. The starting abundance for each species i in the community was drawn randomly from a normal distribution with mean equal to half the equilibrium population density (from μ_∞) for species i and standard deviation equal to $\pm 5\%$ of the mean. We simulated 200 time steps and dropped the first 100 time steps to remove initial condition effects. We then used LAMBDA to estimate the parameters (**A**, **B**) from the simulated datasets, as well as the stability properties.

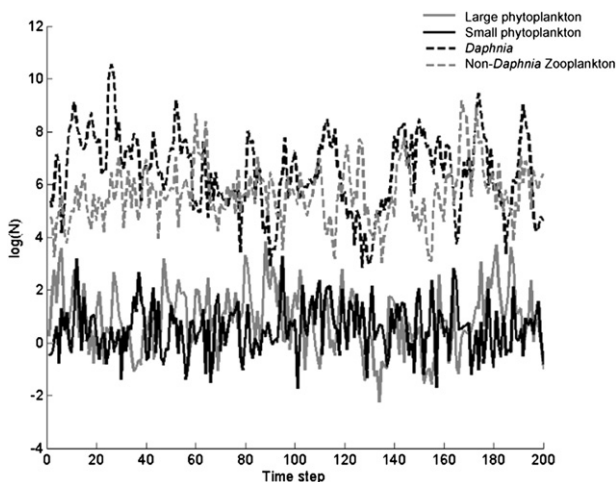


Fig. 1. An example simulation of the MAR(1) process in a four-species community, using the Peter Lake parameters. See text for details.

Table 1

Parameter values for the interactions between the members of the sample community. The four strongest interactions have been highlighted in boldface text.

| | Lg. Phyto. | Sm. Phyto. | <i>Daphnia</i> | Non- <i>Daphnia</i> |
|--|--------------------|---------------------|--------------------|---------------------|
| (A) The known interactions, as programmed in the simulation. These are taken from the "low planktivory" case in Ives et al. (2003) | | | | |
| Lg. Phyto. | 0.50 | -0.36 | 0 | 0 |
| Sm. Phyto. | 0 | 0.07 | -0.02 | -0.10 |
| <i>Daphnia</i> | 0 | 0 | 0.76 | 0 |
| Non- <i>Daphnia</i> | 0.10 | 0 | 0 | 0.56 |
| (B) Estimated interactions (± 1 SD) using the CLS technique on 100 simulated datasets | | | | |
| Lg. Phyto. | 0.45 (0.09) | -0.41 (0.12) | 0 (0.05) | 0 (0.05) |
| Sm. Phyto. | 0 (0.06) | 0.02 (0.10) | -0.01 (0.04) | -0.05 (0.06) |
| <i>Daphnia</i> | 0 (0.12) | 0 (0.11) | 0.71 (0.09) | 0 (0.07) |
| Non- <i>Daphnia</i> | 0 (0.09) | 0.07 (0.17) | -0.01 (0.07) | 0.51 (0.08) |
| (C) Estimated interactions (± 1 SD) using the simplex ML technique on 100 simulated datasets | | | | |
| Lg. Phyto. | 0.43 (0.13) | -0.35 (0.21) | 0.02 (0.09) | 0.03 (0.09) |
| Sm. Phyto. | 0.02 (0.8) | 0.02 (0.13) | 0.01 (0.07) | -0.02 (0.09) |
| <i>Daphnia</i> | 0.02 (0.18) | 0.06 (0.22) | 0.71 (0.19) | 0.07 (0.17) |
| Non- <i>Daphnia</i> | 0.05 (0.16) | 0.18 (0.32) | 0.07 (0.15) | 0.58 (0.22) |
| (D) Estimated interactions (± 1 SD) using the simulated annealing ML technique on 100 simulated datasets | | | | |
| Lg. Phyto. | 0.44 (0.10) | -0.41 (0.12) | 0 (0.06) | -0.1 (0.08) |
| Sm. Phyto. | 0 (0.07) | 0.01 (0.10) | -0.01 (0.06) | -0.06 (0.06) |
| <i>Daphnia</i> | 0.02 (0.14) | 0.01 (0.17) | 0.71 (0.09) | 0 (0.01) |
| Non- <i>Daphnia</i> | -0.02 (0.11) | 0.11 (0.19) | -0.01 (0.08) | 0.51 (0.09) |

To compare the performance across model size, we used the **A**, **B**, and **C** matrices from a 14-species system in Lake Washington (Hampton and Schindler, 2006). Our simulations were performed as above with 100 replicate datasets, and parameters and stability properties estimated for each simulation replicate. We then re-ran the estimation steps on the same simulated data, but successively removed one species. Each time a species was removed, we performed the parameter and stability estimates again. The results were compared across the estimation methods to determine how the number of species in the model affects performance.

Although we use only these two datasets to demonstrate LAMBDA's capabilities, the toolkit can be used to for many other community analyses. For example, Hampton et al. (2008) used LAMBDA to study changes to the trophic structure of Lake Baikal.

5. Results

All estimation methods (CLS, simplex ML, and annealing ML) gave good average estimates of the model parameters for the Peter Lake species interaction matrix **B** (Table 1). The true underlying interaction matrix had three strong positive interactions and one strong negative interaction (Table 1A). These same four interactions

Table 2

Stability properties of the system. The variance in the stationary distribution relative to process error was calculated as $\det(\mathbf{B})^{2/p}$. Return rates were calculated as $\max(\lambda_{\mathbf{B}})$ and $\max(\lambda_{\mathbf{B} \otimes \mathbf{B}})$; reactivity was calculated as $-\text{tr}[\Sigma]/\text{tr}[\mathbf{V}_\infty]$ and $\max(\lambda_{\mathbf{B}'\mathbf{B}}) - 1$. Stability properties were estimated using conditional least squares (CLS), or maximum-likelihood with either simplex search (ML-S) or simulated annealing (ML-A). Estimates are reported as the mean (± 1 S.D.) across 100 simulated replicates.

| | True | CLS Estimate | ML-S Estimate | ML-A Estimate |
|---|-------|--------------|---------------|---------------|
| Variance: $\det(\mathbf{B})^{2/p}$ | 0.12 | 0.13 (0.06) | 0.14 (0.06) | 0.12 (0.06) |
| Return Rate of μ : $\max(\lambda_{\mathbf{B}})$ | 0.76 | 0.72 (0.07) | 0.81 (0.13) | 0.73 (0.07) |
| Return Rate of \mathbf{V} : $\max(\lambda_{\mathbf{B} \otimes \mathbf{B}})$ | 0.57 | 0.53 (0.10) | 0.66 (0.21) | 0.53 (0.11) |
| Reactivity: $-\text{tr}[\Sigma]/\text{tr}[\mathbf{V}_\infty]$ | -0.60 | -0.60 (0.07) | -0.45 (0.28) | -0.59 (0.08) |
| Reactivity: $\max(\lambda_{\mathbf{B}'\mathbf{B}}) - 1$ | -0.43 | -0.38 (0.13) | -0.04 (0.38) | -0.32 (0.13) |

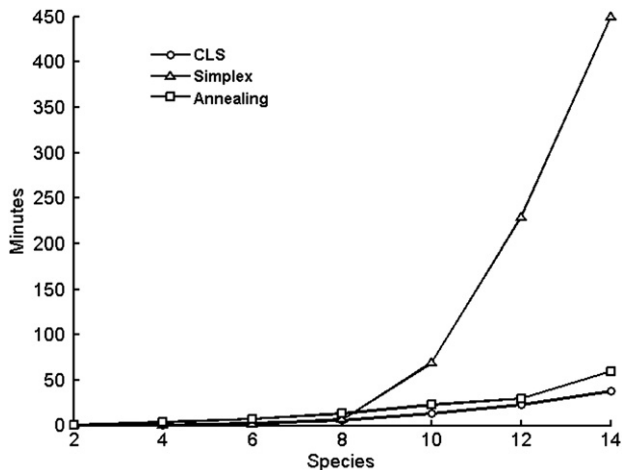


Fig. 2. A comparison of performance (time taken to conduct identical analyses) for the different estimation methods with increasing numbers of variates (simulated species). See text for details.

were identified as the strongest by all three estimation techniques, and their magnitudes were estimated to within $\pm 14\%$ (Table 1B–D). The CLS and annealing ML techniques produced nearly identical results (Table 1B and D). The simplex ML method produced similar mean parameter estimates, but its variance was much higher (Table 1C). The CLS estimation method gave the most accurate estimates of all five stability properties (Table 2). Simplex ML estimation produced extremely high variance on the estimates of reactivity and return rate (Table 2).

Computation time using the CLS method on the Lake Washington dataset increased linearly with the number of species p (on the order of $3p$); using the annealing computation time increased on the order of p^2 ; and using the simplex method computation time increased on the order of $p^3 + p^2$ (Fig. 2).

6. Discussion

This paper demonstrates how our software package, LAMBDA, can analyze multivariate time series data to estimate community interactions and stability using powerful statistical algorithms (Ives et al., 2003). With plankton communities as our example, we illustrate that using LAMBDA, community interaction matrices and stability properties can be successfully estimated (Tables 1 and 2). The plankton communities were used to illustrate LAMBDA performance, but obviously, the results will always depend on community size and time series length.

Based on our simulations, CLS regression is the fastest (Fig. 2) and most accurate (Tables 1 and 2) estimation method. On the other hand, the Simplex method was entirely unsatisfactory: it took far longer (Fig. 2) and produced more variable estimates (Tables 1 and 2) than the other methods. If one desires to use ML methods, we therefore recommend the simulated annealing method.

Estimating community interactions and stability are central to the study and understanding of community dynamics (EPAP, 1999; Wootton and Emmerson, 2005). The MAR(1) model developed by Ives et al. (2003) provides a statistical framework for estimating the

community interaction matrices as well as a variety of community stability properties. MAR(1) models can thus be used with the types of dataset to which ecologists and managers most commonly have access – long term abundance time series. In addition, the MAR(1) framework allows ecologists to study how environmental covariates drive the community dynamics. Recently, MAR(1) models have been used in this way to understand the mechanisms by which nutrient loading and climate change have affected freshwater plankton communities (Hampton et al., 2008, 2006).

Although the MAR(1) model is a powerful tool, it has limitations from which LAMBDA is not immune. For example, LAMBDA calculates a single stationary distribution, but a system may have more than one stable state (Ives et al., 2008). Additionally, key environmental drivers must be carefully identified, because environmentally-driven synchrony can alter \mathbf{B} estimates (Mutshinda et al., 2009). Finally, although we do not address observation error directly here, it clearly can have an impact (Wootton and Emmerson, 2005), and should be quantified to the extent possible.

Because of their generality, and because they can make use of commonly available datasets, MAR(1) techniques can be applied to many systems, provided one has the computational tools to perform the analysis. The LAMBDA software package provides these tools. With LAMBDA, ecologists have a toolbox of powerful statistical tools for analyzing multi-species time series data.

Acknowledgments

We are grateful to the many people who assisted in LAMBDA's development, including A. Ives, B. Dennis, S. Hampton, S. Katz, M. Scheuerell, and three anonymous reviewers, as well as participants of two workshops (at NCEAS and the 2007 ESA meeting) who provided comments and testing. SVV's funding was provided through an NRC Post-doctoral Research Apprenticeship.

References

- Cerny, V., 1985. A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications* 45, 41–51.
- EPAP, 1999. Ecosystem-based fishery management. Ecosystem Principles Advisory Panel, Washington, D.C., pp. 54.
- Hampton, S.E., Izmet'eva, L.R., Moore, M.V., Katz, S.L., Dennis, B., Silow, E.A., 2008. Sixty years of environmental change in the world's largest lake – Lake Baikal, Siberia. *Global Change Biology* 14, 1947–1958.
- Hampton, S.E., Scheuerell, M.D., Schindler, D.E., 2006. Coalescence in the Lake Washington story: interaction strengths in a planktonic food web. *Limnology and Oceanography* 51, 2042–2051.
- Hampton, S.E., Schindler, D.E., 2006. Empirical evaluation of observation scale effects in community time series. *Oikos* 113, 424–439.
- Ives, A.R., Einarsson, A., Jansen, V.A.A., Gardarsson, A., 2008. High-amplitude fluctuations and alternative dynamical states of midges in Lake Myvatn. *Nature* 452, 84–87.
- Ives, A.R., Dennis, B., Cottingham, K.L., Carpenter, S.R., 2003. Estimating community stability and ecological interactions from time series data. *Ecological Monographs* 73, 301–330.
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. *Science* 220, 671–680.
- Mathworks, T., 2005. MATLAB. In: Release, fourteenth ed. The Mathworks Inc, Natick, MA.
- Mutshinda, C.M., O'Hara, R.B., Woiwod, I.P., 2009. What drives community dynamics? *Proceedings of the Royal Society B: Biological Sciences* 276, 2923–2929.
- Wootton, J.T., Emmerson, M., 2005. Measurement of interaction strength in nature. *Annual Review of Ecology, Evolution, and Systematics* 36, 419–444.