Automated modeling of phytoplankton growth using ecological domain knowledge

L. Todorovski*, S. Džeroski** & B. Kompare***
*University of Ljubljana, Faculty of Medicine, Vrazov trg 2, 1105 Ljubljana, Slovenia
**Jožef Stefan Institute, Jamova 39, 1111 Ljubljana, Slovenia
***University of Ljubljana, Faculty of Civil and Geodetic Engineering, Hajdrihova 28, 1001 Ljubljana, Slovenia

Abstract

Using ecological domain knowledge, machine discovery systems can help human experts to generate models from ıneasured data. In contrast with traditional modeling methods, which are used to identify parameter values of the model with prescribed structure, machine learning tools identify the structure of the model as well.

In the paper, we present LAGRAMGE, an equation discovery system that uses context free grammars to define the space of possible model structures, and can also make use of domain specific background knowledge in the form of function definitions. We use LAGRAMGE to automate the modeling of phytoplankton growth in Lake Glumsoe, Denmark. The structure of the automaticly constructed model agrees with human experts expectations. The model can be successfully used for short-term prediction of the phytoplankton concentration.

1 Introduction

The task of modeling dynamic systems is to find a model that describes an observed behavior. Usually, a model of a dynamical system is a set of differential equations that specify the change of system variables over time.

Mainstream system identification methods work under the assumption that the model structure, i.e. the form of the differential equations, is known [6]. The task is then to determine the values of the constant parameters in the equations, so that the model fits measured data. Machine discovery systems, such as LAGRANGE [2] and GoldHorn [5] do not assume a prescribed model structure, but rather explore a space of (possibly nonlinear) equations. They help human experts to identify the structure of the model as well as the values of the constant parameters.

Machine discovery systems can be used for automated modeling of ecological dynamic systems. Kompare [4] used LAGRANGE and GoldHorn to produce a model for predicting algal growth in the Lagoon of Venice. Several problems arise when using these systems for modeling experimental data. LAGRANGE discovered some equations resembling the optimal temperature for algal growth, but no good equations were discovered, from the viewpoint of what human experts expected. The cause for this was the level of noise in the data. GoldHorn incorporates methods for discovery from noisy data, so the expected equations were discovered, but a lot of equations with nonacceptable structure were ranked as better fitted.

These problems led to the idea of narrowing the search space of equations, that LAGRANGE and GoldHorn consider in the process of discovery. In the area of machine learning, the concept of declarative language bias [7] is used to specify the hypothesis space (space of all possible equations, in the task of equation discovery). It was observed that smaller hypothesis space would lead to better performance of the learned concept (model) on a test set of unseen cases.

In the paper, we present an equation discovery system LAGRAMGE, that uses context free grammars as a formalism for declarative bias. The grammar can use the usual mathematical operators defined in the C programming language as well as additional functions defined by the grammar at hand. The grammar is specified according to the domain specific knowledge, and focuses the equation discovery process to equations with acceptable structure within the domain of use.

LAGRAMGE was used on the problem of modeling the phytoplankon growth in the Lake Glumsoe in Denmark. The structure of equations discovered make sense from the ecological point of view. They can also be used as accurate short-term predictors for phytoplankton growth. The performance of the predictor is comparable to the performances of no-change and same-change predictors for prediction period of one to five days.

The paper is organized as follows. Section 2 gives overview of LAGRAMGE. First it defines the equation discovery problem, as addressed by LAGRAMGE. The use of grammars for incorporating domain specific knowledge in the equation discovery is shown on an example of a simple ecological domain. It ends with a brief description of the algorithm. In Section 3 the Lake Glumsoe domain is presented. Section 4 describes the

performed experiments in modeling Lake Glumsoe and the evaluation of the obtained model. Finally, Section 5 concludes with a summary of the results.

2 The equation discovery system LAGRAMGE

2.1 Problem definition

The problem of equation discovery, as addressed by LAGRAMGE, can be defined as follows.
Given:

- context free grammar $G = (N, T, P, S)$ (see Section 2.2) and
- input data $D = (V, v_d, M)$, where
  - $V = \{v_1, v_2, \ldots v_n\}$ is a set of domain variables,
  - $v_d \in V$ is dependent variable and
  - M is a tuple of one or more measurement sets. Each mesuarement set is a table of measurements of the domain variables in distinct time points:

time	v_1	v_2	$\ldots$	v_n
t_0	$v_{1,0}$	$v_{2,0}$	$\ldots$	$v_{n,0}$
t_1	$v_{1,1}$	$v_{2,1}$	$\ldots$	$v_{n,1}$
t_2	$v_{1,2}$	$v_{2,2}$	$\ldots$	$v_{n,2}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
t_N	$v_{1,N}$	$v_{2,N}$	$\ldots$	$v_{n,N}$

find an equation for expressing the dependent variable v_d in terms of variables in V. This equation is expected to minimize the discrepance between the measured and calculated values of the dependent variable. The equation can be:

- differential, i.e. of the form $\partial v_d / \partial t = E$, or
- ordinary, i.e. of the form $v_d = E$,

where E is an expression that can be derived from the context free grammar G.

2.2 The declarative bias formalism

The syntax of the expressions on the right side of the equation are prescribed with the context free grammar $G = (N, T, P, S)$. N, T and P are sets of nonterminals, terminals and productions and $S \in N$ is the starting nonterminal. Productions in P are of the form $A \rightarrow \alpha$, where $A \in N$ is called the left side and $\alpha \in (N \cup T)^*$ the right side of the production.

The grammar used to describe the declarative bias for equation finding has several symbols with special meanings. The terminal *const* $\in T$ is used to denote a constant parameter in an equation, that has to be fitted to the input data. The terminals v_i are used to denote variables from the input domain D. Finally, the nonterminal $v \in N$ denotes any variable from the input domain. Productions connecting this symbol to the terminals v_i are attached to v automatically, i.e. $\forall v_i \in V : v \rightarrow v_i \in P$.

The only restriction on the grammar G is that it has to generate expressions that are legal in the C programming language. This means that it can use all C built-in operators and functions. Additional functions, representing background knowledge about the addressed domain, can be used as long as they are defined in conjunction with the grammar.

Expressions can be derived in grammar G from nonterminal symbol S with applying productions in P. We use a production from P to expand the nonterminal on its left side with the symbols on its right side. Starting with expression S, we expand it with productions from P, until it is composed of terminals only.

2.3 An example - aquatic ecosystem

We illustrate the use of grammars on a simple aquatic ecosystem domain. A system of differential equations describes the evolution of the concentrations of nutrient N, phytoplankton P and zooplankton Z in an aquatic environment:

$$\begin{aligned} \dot{N} &= -\frac{NP}{k_N + N} \\ \dot{P} &= \frac{NP}{k_N + N} - r_P P - \frac{PZ}{k_P + P} \\ \dot{Z} &= \frac{PZ}{k_P + P} - r_Z Z. \end{aligned}$$

```
double monod(double c, double v) {
        return(v / (v + c));
}
```

$$
\begin{array}{rcl}
N & = & \{E, F, M, v\} \\
T & = & \{+, const, *, \mathtt{monod}, (, ,,), N, P, Z\} \\
P & = & \left\{ \begin{array}{rcl} E & \rightarrow & const \mid const * F \mid E + const * F \\ F & \rightarrow & v \mid M \mid v * M \\ M & \rightarrow & \mathtt{monod}(const, v) \end{array} \right\} \\
S & = & E
\end{array}
$$

The grammar above is constructed based on background ecological knowledge. The Monod term is defined with function `monod` in the C programming language and incorporated in the grammar through nonterminal M. The rest of the grammar is used to combine the Monod terms with domain variables in legal C expressions. We can derive the expression $const * N/(const + N)$ from the grammar with the following derivation:

Expression	Production used
$E \rightarrow$	$E \rightarrow const * F$
$const * F \rightarrow$	$F \rightarrow M$
$const * M \rightarrow$	$M \rightarrow \mathtt{monod}(const, v)$
$const * \mathtt{monod}(const, v) \rightarrow$	$v \rightarrow N$
$const * \mathtt{monod}(const, N)$	

2.4 LAGRAMGE - the algorithm

Expressions generated by the context free grammar G contain one or more special terminal symbols *const*. A nonlinear fitting method is applied to determine the values of these parameters. The fitting method minimizes the value of the error function $Error(\mathbf{c})$, i.e. if $\mathbf{c}$ is the vector of constant parameters in expression E, then the result of the fitting algorithm is a vector of parameter values $\mathbf{c}^*$, such that $\mathbf{c}^* = \mathrm{argmin}_{\mathbf{c} \in R^{n_c}} \{Error(\mathbf{c})\}$. The error function *Error* is a sum of squared errors function, defined in the following manner:

- for differential equation of the form $\partial v_d / \partial t = E$:

  $Error(\mathbf{c}) = \sum_{i=0}^{N} \left[v_{d,i} - \left(v_{d,0} + \int_{t_0}^{t_i} E(\mathbf{c}, v_1, \ldots v_n) \right) \right]^2$, and

- for ordinary equation of the form $v_d = E$:

  $Error(\mathbf{c}) = \sum_{i=0}^{N} (v_{d,i} - E(\mathbf{c}, v_{1,i}, \ldots v_{d-1,i}, v_{d+1,i}, \ldots v_{n,i}))^2$.

The downhill simplex and Levenberg-Marquart algorithms [8] can be used for minimization of the error function.

Furthermore, the value of a heuristic function of the expression is evaluated. It is based on the sum of squared errors value *SSE* calculated by the fitting method ($SSE(E) = Error(\mathbf{c}^*)$). An alternative heuristic function *MDL* can be used, that take into account the length l of E:

$$MDL(E) = SSE(E) + \frac{l}{10 \cdot l_{max}} \sigma_{v_d},$$

where l_{max} is the length of the largest expression generated by the grammar. The *MDL* (minimal description length) heuristic function prefers shorter equations.

The LAGRAMGE algorithm searches for the best equation according to the heuristic function. A beam search procedure is used to search the space of all equations that can be derived by the context free grammar G.

3 Lake Glumsoe

The Lake Glumsoe [3] is situated in a sub-glacial valley in Denmark. It is shallow with average depth of about 2 m and its surface area is 266,000 m^2. For several years, it has received mechanically-biologically treated waste water from a community with 3,000 inhabitants and mainly agricultural (almost no industry) surrounding. The high nitrogen and phosphorus concentration in the waste water caused hypereutrophication. The lake contains no submerged vegetation, probably due to the low transparency of the water and oxygen deficit at the bottom of the lake.

Concentrations of phytoplankton (*phyt*), zooplankton (*zoo*), soluble nitrogen (*NS*) and soluble phosphorus (*PS*) were considered relevant for modeling the phytoplankton growth. State variables were measured in 14 distinct time points, over a period of two months. Due to the small amount of measured data, additional processing was applied to obtain experimental data [4, 1]. First, dotted graphs of measured data were plotted and given to three human experts to draw a curve that, in their own opinion, describes the dynamic behavior of the observed state variable. Curves drawn by the human experts were then smoothed with Besier splines. Finally, three experimental data sets were obtained by sampling the splines from each human expert at regular time intervals with time step $h = 0.1$ day. The dynamic behaviors of the phytoplankton in three experimental data sets are shown on Figure 1.

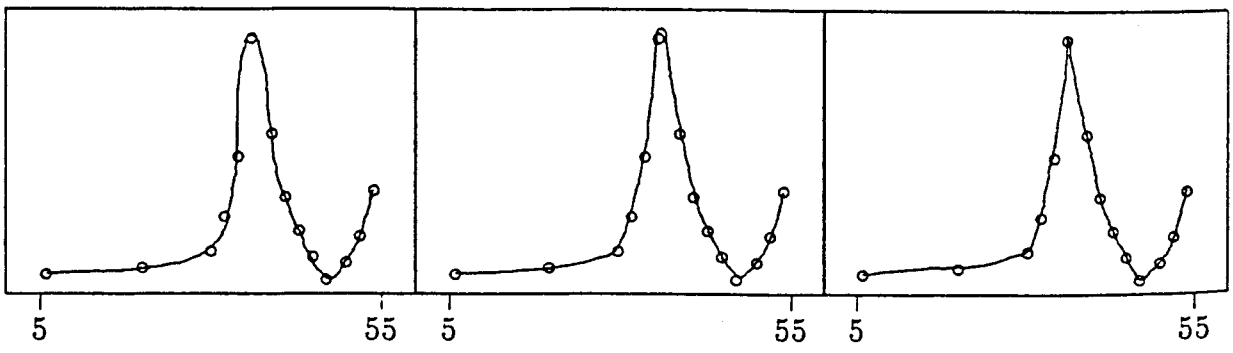

Figure 1: Phytoplankton growth as seen by three domain experts

4 Experiments

The grammar used in the experiments was constructed based on ecological background knowledge on algal growth. Phosphorus and nitrogen are nutrients for phytoplankton and can thus appear in Monod terms (productions for nonterminals M and VM). Other terms model the decay of phytoplankton and the feeding of zooplankton on phytoplankton. 72 equations can be derived from the grammar.

There are some constraints on the values of the parameters in the equations specified by the grammar: they have to be positive, except for the ones in front of the decay term and the term describing the feeding of zooplankton on phytoplankton.

```
double monod(double c, double v) {
       return(v / (v + c));
}
```

$$
\begin{array}{lll}
E & \rightarrow & F * v\_phyt + const * v\_phyt \\
 & | & F * v\_phyt + const * v\_phyt + const * V\_phyt * v\_zoo \\
F & \rightarrow & const * M * M \mid const * M + const * M \mid const * M \\
M & \rightarrow & \text{monod}(const, VM) \\
VM & \rightarrow & v\_PS \mid v\_NS
\end{array}
$$

In the experiments we used the 'leave one out' testing method: LAGRAMGE was given two sets of data for equation discovery, and the best equation discovered was then tested on the remaining data set. The equation was tested on the task of predicting phytoplankton growth.

In experiments with MDL heuristic (all possible 72 equations were considered), the best equation discovered by LAGRAMGE was chosen that satisfied the constraints for the parameters' values. The three equations obtained have the same structure:

$$\dot{phyt} = const_1 * phyt * PS/(const_2 + PS) + const_3 * phyt$$

The structure of the equations discovered makes sense from an ecological point of view. It tells us that phosphorus is a limiting factor for phytoplankton growth in the lake.

The constant parameters' values, as well as the correlation coefficients between the measured and predicted values of phytoplankton (on the testing set) for each of the three equations are shown below:

Training data sets	$const_1$	$const_2$	$const_3$	r
1, 2	0.616791	0.101413	-0.442205	0.999452
1, 3	0.762913	0.0796594	-0.591642	0.998856
2, 3	0.3831	0.444443	-0.155398	0.99958

It can be seen that all equations give accurate short term predictions for phytoplankton growth. Furthermore, we tested the robustness of the predictor on increasing the prediction period. The summary of the results (correlation coefficients between measured and predicted values) for prediction periods of one, two and five days are given bellow:

Training data sets	1 day	2 days	5 days
1, 2	0.983556	0.941354	0.724288
1, 3	0.984942	0.955216	0.813708
2, 3	0.985263	0.956632	0.726655

Finally, we compared the accuracy of the obtained predictor with the accuracies of two simple predictors: no-change and same-change. The no-change predictor predicts that the value of the variable in the next time point will be the same as the present value ($phyt(t+h) = phyt(t)$). Same-change predicts the same change, as the change in previous time step ($phyt(t+h) - phyt(t) = phyt(t) - phyt(t-h)$). The graphs on Figure 2 show the dependence of correlation coefficients between the measured values and values predicted with the three different predictors on increasing prediction period for all experimental data sets.

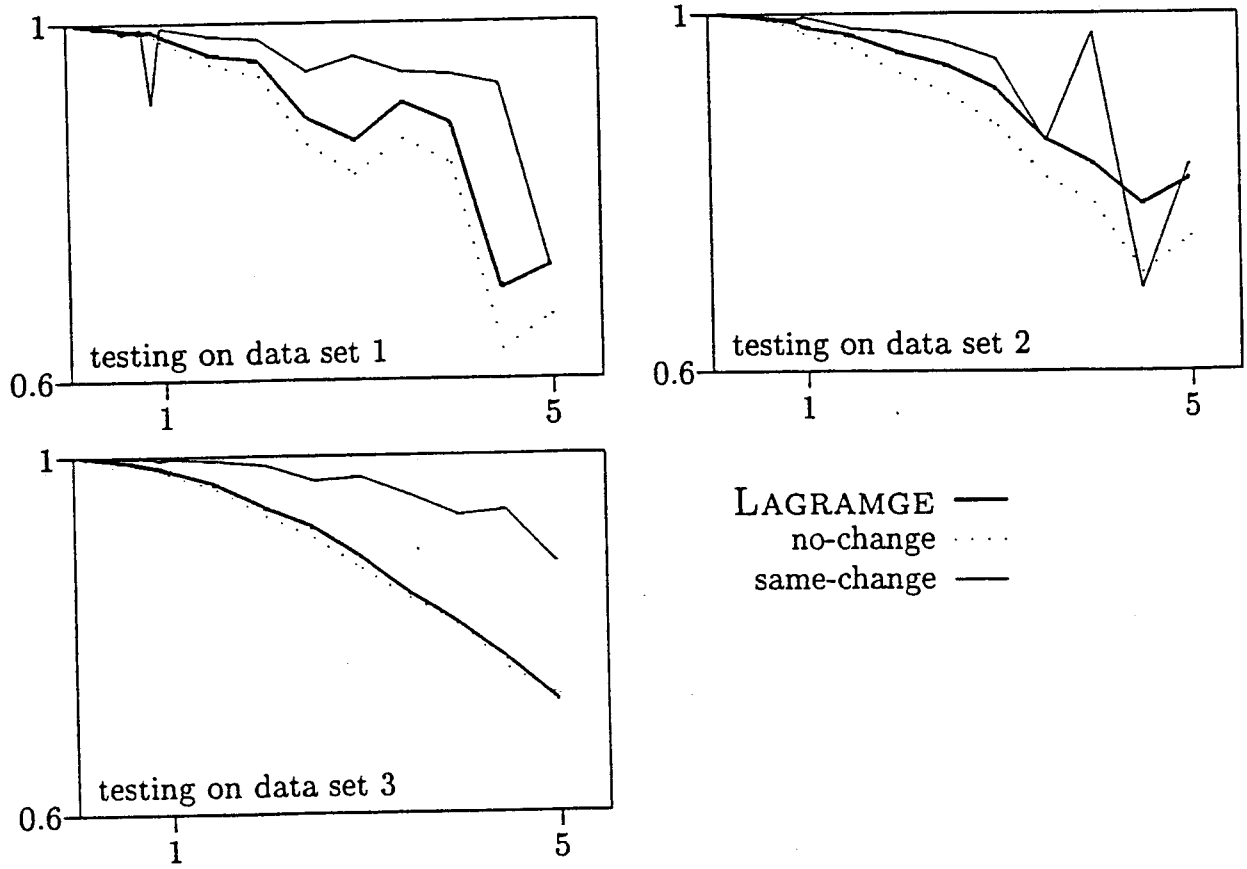

Figure 2: The correlation coefficient between the measured and predicted values as the prediction time increases.

The graphs show that the accuracy of the predictions decreases as the prediction time increases, which was expected. The performance and robustness of all predictors is comparable. Same-change predictor has better

performance than the one obtained with LAGRAMGE, especially on the third data set. The no-change predictor has the lowest accuracy on all data sets.

5 Discussion

We have presented an equation discovery system LAGRAMGE that uses declarative bias to incorporate domain specific knowledge in the process of equation discovery. Context free grammars are used to specify the equation structure. Background knowledge in the form of function definitions can also be used. LAGRAMGE (heuristically) searches the space of equation structures defined by a grammar. LAGRAMGE uses nonlinear optimization procedures, such as downhill simplex and Levenberg-Marquardt to fit equation parameters. The search heuristic used is based on the fit, but can also take into account the length of equations (*MDL*). LAGRAMGE can find both ordinary and differential equations, in both implicit and explicit form. It can also take into account more than one behavior of a dynamic system.

We used LAGRAMGE on the task of modeling phytoplankton growth in the Lake Glumsoe, Denmark. For three different learning data sets we obtained equations with the same structure, which is acceptable in the sense of ecological expert's knowledge. The structure of the equations tells us that phosphorus is a limiting factor for phytoplankton growth in the lake.

Furthermore, equations obtained with LAGRAMGE were tested on the problem of the prediction of phytoplankton growth. Tests shown that the equations obtained can be used for accurate short-term (one or two days ahead) predictions. The accuracy of the predictor is comparable to the simple no-change and same-change predictors. We expect that the accuracy and robustness of the predictor, obtained with LAGRAMGE, can be improved with providing better measurement data to the process of equation discovery. Namely, the available experimental data were obtained on the basis of only 14 measurements made in a period of two months. Extensive regular measurements can be done over one year period to provide good basis for automated modeling.

Acknowledgements

This work was supported in part by the Slovenian Ministry of Science and Technology under the project *ILP for Knowledge Discovery in Ecological Databases* and by the ESPRIT IV project *20237 ILP2*.

References

[1] Demšar, D. (1996). Experiments in automated modeling of ecological processes in Lake Glumsoe. BSc Thesis, Faculty of Computer and Information Science, University of Ljubljana, Slovenia. In Slovenian.

[2] Džeroski, S. and Todorovski, L. (1993). Discovering dynamics. In *Proc. Tenth International Conference on Machine Learning*, pages 97–103. Morgan Kaufmann, San Mateo, CA, 1993.

[3] Joergensen, S., Kamp-Nielsen, L., Chirstensen, T., Windolf-Nielsen, J., and Westergaard, B. (1986). Validation of a prognosis based upon a eutrophication model. *Ecological Modelling*, 32:165–182.

[4] Kompare, B. (1995). The use of artificial intelligence in ecological modeling. PhD Thesis, Royal Danish School of Pharmacy, Copenhagen, Denmark.

[5] Križman, V. (1994). Handling noisy data in automated modeling of dynamical systems. MSc Thesis, Faculty of Computer and Information Science, University of Ljubljana, Slovenia.

[6] Ljung, L. (1993). Modelling of industrial systems. In *Proc. Seventh International Symposium on Methodologies for Intelligent Systems*, pages 338–349. Springer, Berlin, 1993.

[7] Nédellec, C., Rouveirol, C., Adé, H., Bergadano, F., and Tausend, B. (1996). Declarative bias in ILP. In De Raedt, L., editor, *Advances in Inductive Logic Programming*, pages 82–103. IOS press.

[8] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1986). *Numerical Recipes*. Cambridge University Press, Cambridge, MA.