SELF-ORGANIZING DATA MINING BASED ON GMDH PRINCIPLE

Dipl.-Ing. Frank Lemke*, Prof. Dr.rer.oec.habil. Johann-Adolf Müller**
*DeltaDesign Software, Bergener Str.1, D-10439 Berlin Germany
Phone: +49-30- 4443585, email: knowledgeminer@iworld.to
**HTW Dresden, Fachbereich Informatik/Mathematik
F.-List-Platz 1, Dresden D-01069 Germany
Phone: +49-351-4623322, Fax:: +49-351-4623671
email: muellerj@informatik.htw-dresden.de

ABSTRACT: Most important for a more sophisticated data mining is to limit the involvement of users in the overall modeling process to the inclusion of existing a priori knowledge while making this process more automated and more objective. Self-organizing data mining introduces principles of evolution - inheritance, mutation and selection - for generating a network structure systematically enabling automatic model structure synthesis and model validation. „KnowledgeMiner“ was designed to support the knowledge extraction process on a highly automated level. Implemented are 3 different GMDH-type self-organizing modeling algorithms at present: GMDH, Analog Complexing and Fuzzy rule induction using GMDH to make knowledge extraction systematically, fast, successful and easy-to-use even for large and complex systems.

KEYWORDS: Self-organizing modeling, data mining, financial analysis, prediction, neural networks, fuzzy modeling

1. SELF-ORGANIZING DATA MINING

Today, there is an increased need to discover information - contextual data - non obvious and valuable for decision making from a large collection of data efficiently. This is an interactive and iterative process of various subtasks and decisions and is called Knowledge Discovery [Fayyad (1996)]. The engine of Knowledge Discovery - where data is transformed into knowledge for decision making - is Data Mining.
There are very different data mining tools available and many papers are published describing data mining techniques. Most important for a more sophisticated data mining is to try to limit the involvement of users in the entire data mining process to the inclusion of well-known a priori knowledge, exclusively, while making this process more automated and more objective. Most users' primary interest is in model results proper without having to have extensive knowledge of mathematical, cybernetic and statistical techniques or sufficient time for dialog driven modeling tools. Soft computing, i.e., Fuzzy Modeling, Neural Networks, Genetic Algorithms and other methods of automatic model generation, is a way to mine data by generating mathematical models from empirical data more or less automatic.
In the past years there has been much publicity about the ability of Artificial Neural Networks to learn and to generalize [Bigus (1996)] despite important problems with design, development and application of Neural Networks [Müller (1998)]:

- Neural Networks have no explanatory power by default, that is, describing why results are as they are. This means that the knowledge (models) extracted by Neural Networks is still hidden and distributed over the network.
- There is no systematical approach for designing and developing Neural Networks. It is a trial-and-error process.
- Training of Neural Networks is a kind of statistical estimation often using algorithms that are slower and less effective than algorithms used in statistical software.
- If noise is considerable in a data sample the generated models systematically tend being over-fitted.

In contrast to Neural Networks that use [Kingdon (1997)]

- Genetic Algorithms as an external procedure to optimize the network architecture and
- several pruning techniques to counteract over-training,

self-organizing data mining based on Group Method of Data Handling (GMDH) introduces principles of evolution - inheritance, mutation and selection - for generating a network structure systematically enabling automatic model structure synthesis and model validation.

2. GROUP METHOD OF DATA HANDLING (GMDH)

The traditional GMDH algorithm was developed by A.G. Ivakhnenko in 1967. This approach was described by Madala and Ivakhnenko [Madala (1994)]. Further development of GMDH algorithm in connection with application of cross-validation principles, optimization of the structure of transfer functions (neurons) and generation of systems of equations a/o. was realized by Lemke [Lemke (1997)]. Fig. 1 illustrates the creation of such a model of optimal complexity.

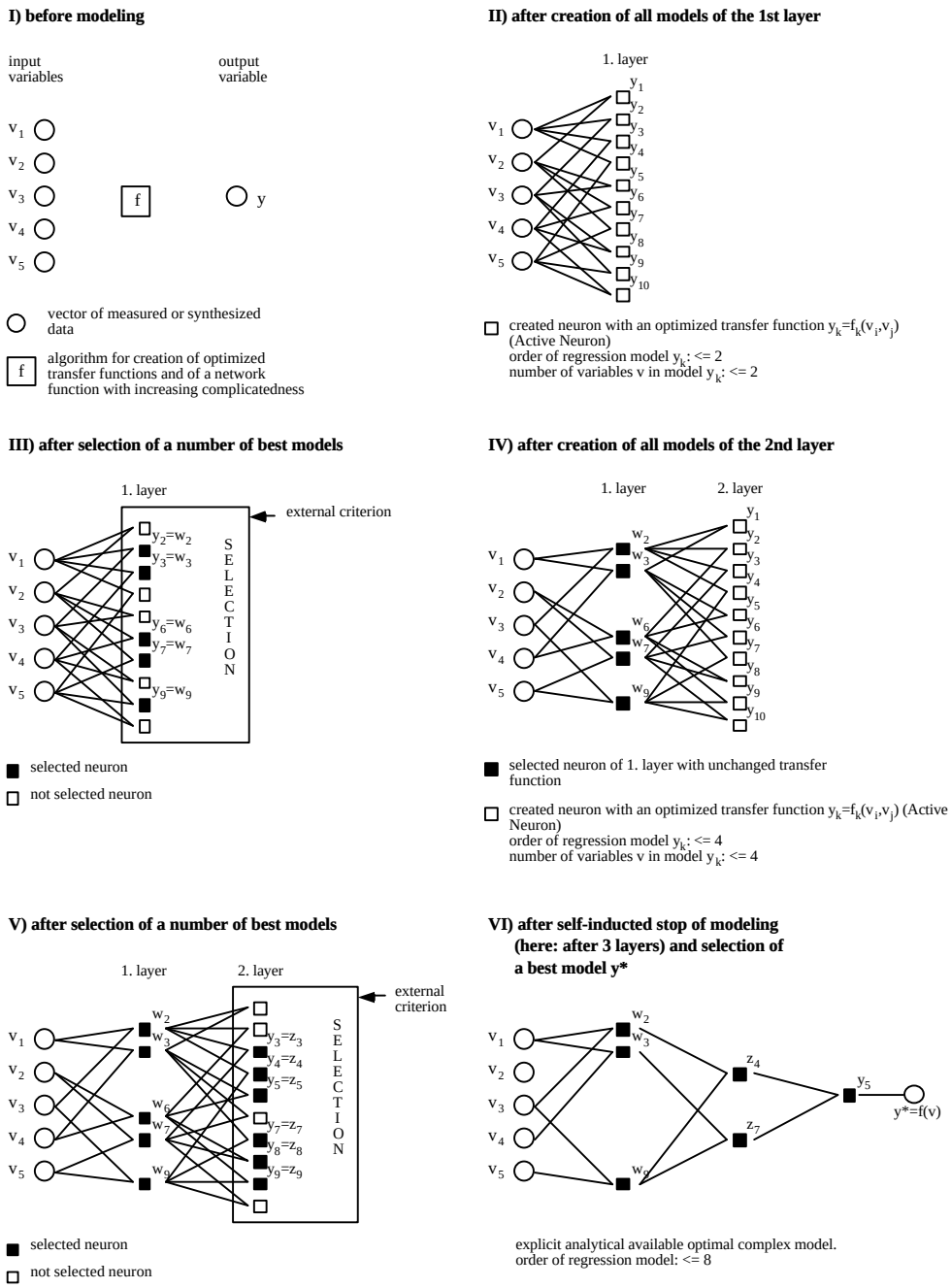

Figure 1: Creation of an optimal complex model using a multi-layered GMDH algorithm [Lemke (1997)]

The scientific foundation of the theory of GMDH modeling is a combination of the two basic foundations of Neural Network modeling

- the black-box method as a principal approach to analyze systems on the basis of input-output samples;
- the connectionism as a representation for complex functions through networks of elementary functions

with the following three principles [Müller (1998)]:
- the cybernetic principle of self-organization, which means an adaptive creation of a network without subjective points given;
- the principle of external complement, which enables an objective choice of a model of optimal complexity;
- the principle of regularization of ill-posed tasks.

Models are generated adaptively from data in form of networks of active neurons in an evolutionary fashion of repetitive generation of populations of competing models of growing complexity, their validation and selection until an optimal complex model - not too simple and not too complex - have been created. That is, growing a tree-like network out of seed information (input and output variables' data) in an evolutionary fashion of pair-wise combination and survival-of-the-fittest selection from a simple single individual (neuron) to a desired final, not overspecialized behavior (model). Neither, the number of neurons and the number of layers in the network, nor the actual behavior of each created neuron is predefined. All this is adjusting during the process of self-organization, and therefore, is called self-organizing data mining.

3. „KNOWLEDGEMINER“ - A SELF-ORGANIZING MODELING SOFTWARE TOOL

„KnowledgeMiner“ (http://www.scriptsoftware.com/km/index.html) is a powerful and easy-to-use modeling tool which was designed to support the knowledge extraction process (table I) on a highly automated level and which has implemented three advanced self-organizing modeling technologies at present: GMDH (sect.3.1), Analog Complexing (AC) (sect.3.2) and Fuzzy rule induction using GMDH (Fuzzy-GMDH) (sect.3.3). Objective Cluster Analysis (OCA) [Müller (1998)] will be realized in the future.

Data Mining functions	*Algorithm*
classification	OCA, GMDH, Fuzzy-GMDH, AC
clustering	OCA
modeling	GMDH, Fuzzy-GMDH
time series forecasting	AC, GMDH, Fuzzy-GMDH, OCA
sequential patterns	AC

Table I:. Algorithms for self-organizing modeling

3.1 GMDH IMPLEMENTATION

„KnowledgeMiner“ has implemented 3 different GMDH-type self-organizing modeling algorithms to make knowledge extraction systematically, fast, successful and easy-to-use even for large and complex systems.

a. Active neurons

„KnowledgeMiner“ performs self-organization of an optimal complex transfer function of each created neuron (active neuron). For many dynamic and static systems, these neurons can have a second order polynomial form

$$f(v_i, v_j) = a_0 + a_1 v_i + a_2 v_j + a_3 v_i v_j + a_4 v_i^2 + a_5 v_j^2 \ .$$

The arguments v_i , v_j represent all forms of input data like non lagged input variables, lagged input variables, derivative input variable or even functions or models.

Beginning at the simplest possible transfer function $f(v_i,v_j)=a_0$, an optimal complex neuron is evolved by repetitive creation, validation and selection of different transfer function representations. One important feature of active neurons is that they are able to select significant input variables themselves.

b. Network synthesis (multi-input/single-output model)

Secondly, an algorithm for self-organization of multi-layered networks of active neurons is implemented. It performs the creation of an optimal complex network structure (optimal number of neurons and number of layers) including cross-validation and selection of a number of best model candidates out of populations of competitive models. The algorithm ensures, for example, that even if creation of nonlinear model was chosen as permissible it really could be possible that a linear model only will be selected as optimal, finally. The implemented selection criterion subdivides the

data set internally into training and testing data sets dynamically. This means, the user doesn't need to process data subdivision in any way; the cross-validation criterion uses virtually the complete data set for training as well as for testing synthesized models. The result of the modeling process is an easily accessible and visible analytical model (model graph, model equation, model data output). All created models are stored in a model base and are immediately applicable for analysis and short- to long-term status-quo or what-if predictions.

c. Systems of equations (multi-input/multi-output model)

One important feature of „KnowledgeMiner“ is self-organization of an optimal, autonomous system of equations. This system has to be free of mathematical conflicts and can be viewed as a network of interconnected GMDH networks which is visible through a system graph and applicable for long-term status-quo prediction of the whole system. It provides the only way to predict a set of input-output models autonomously, objectively and without additional efforts.

Example: Financial forecasting

In result of data selection are given 100 daily close prices of 10 variables of the German stock market (dollar exchange rate, stock rates of BMW, VW, AUDI, Ford, Porsche, stock indexes: DAX, FAZ and financial characteristics: Discount and Lombard rate) for the period of August 5, 1995 through December 11, 1995.

These data (90 observations from August 5, 1995 through November 27, 1995) were used as is to generate models for long-term prediction (10 days: November 28, 1995 through December 11, 1995). Normalization and denormalization of the data as an integrated part of the overall modeling algorithm was processed automatically. After defining the input variables and their maximum dynamics (all 10 variables plus their lagged samples up to a lag of 15 $\rightarrow$ 159 inputs), the output variables (each of the 10 variables) and the type of model (linear system of equations), the complete modeling process runs automatically while creating and validating thousands of different models of increasing complexity. As an optimal model was generated the model table II.

$BMW_t = 59.16 + 0.776\ BMW_{t-1} + 73.632\ Dollar_{t-1} - 0{,}231\ Ford_{t-2} + 0.135 Ford_{t-6} + 0.672 VW_t - 0.472 VW_{t-1}$

$Dollar_t = 0.364 + 0.906\ Dollar_{t-1} - 0.0002\ Ford_{t-2} + 0.0006\ FAZ_t - 0.0005\ FAZ_{t-1} - 0.0003\ FAZ_{t-2} + 0.0001\ FAZ_{t-11}$

$VW_t = -48.728 + 0.87\ VW_{t-1} - 0.091\ VW_{t-3} + 0.078\ DAX_t - 0.03\ DAX_{t-1} - 0.057\ BMW_{t-1} + 0.173\ Audi_{t-9}$

$Audi_t = 10.5 + 0.17\ Audi_{t-2} + 258.51\ Dollar_{t-1} + 109.7\ Dollar_{t-5} - 38.32\ Dollar_{t-8} - 0.03\ DAX_{t-2}$

$FAZ_t = 299.7 + 0.29\ DAX_t - 0.05\ DAX_{t-5} + 0.11\ Porsche_t + 0.06\ BMW_{t-13} - 0.178\ Ford_{t-8}$

$DAX_t = 1420.8 + 0.63\ DAX_{t-1} + 232.3\ Dollar_{t-1} - 0.98\ FAZ_{t-2} + 0.84\ VW_{t-1} - 0.23\ Ford_{t-2} - 0.405\ Ford_{t-7}$

$Discount_t = -2 + Lombard_t$

$Lombard_t = 0.29 + 0.947\ Lombard_{t-1}$

$Ford_t = 770.04 + 0.35\ Ford_{t-3} + 0.18\ Ford_{t-4} - 13.08\ Discount_{t-3} - 57.86\ Lombard_{t-15}$

$Porsche_t = -122.66 + 0.586\ Porsche_{t-1} + 154.8\ Dollar_{t-1} + 0.077\ DAX_{t-15}.$

Table II:. System of equations obtained by "KnowledgeMiner"

3.2 ANALOG COMPLEXING IMPLEMENTATION

„KnowledgeMiner“ provides an Analog Complexing algorithm [Müller (1998)] for prediction of the most fuzzy processes like financial or other markets. It is a multi-dimensional search engine to select most similar, past system states relative to a chosen (actual) reference state. This means, searching for analogous patterns in the data set is usually not only processed on a single time series (column) but on a specified, representative set of time series simultaneously to extract significant hidden knowledge. Additionally, it is possible to let the algorithm search for different pattern length (number of rows a pattern consists of) within one modeling process. All selected patterns, either of the same or different pattern length, are then combined to synthesize a most likely prediction. „KnowledgeMiner“ performs this in an objective way using a GMDH algorithm to find out the optimal number of patterns and their composition to obtain a best result.

Example: Financial forecasting

Based on observations (August 30, 1996 - January 9, 1998) of prices of diverse stocks (the 30 stocks composing the DAX and 23 other national and foreign equities) long-time predictions for all 50 variables are generated by means of GMDH algorithms and Analog Complexing. Table III compares the mean values over all 50 variables and several prediction periods of out-of sample long-time predictions (prediction period T=5, 10, 15, 20, 25, 30, 35, 40). It is shown, that Analog Complexing gives the same or better prediction accuracy whereby the necessary evaluation time is small.

T	GMDH	Analog Complexing
5	0,0276	0,0264
10	0,0403	0,0344
15	0,0505	0,0426
20	0,0608	0,0493
25	0,0698	0,0554
30	0,0834	0,0590
35	0,0851	0,0643
40	0,1211	0,066

Table III:. Comparison of long-time prediction error (MAD [%])

3.3 FUZZY RULE INDUCTION USING GMDH

Fuzzy modeling is an approach to form a system model using a description language based on fuzzy logic with fuzzy predicates. Such a description is able qualitatively to describe a dynamic multi-input/multi- output system by means of a system of fuzzy rules.
This GMDH approach can be used to generate fuzzy models. In the following we suggest considering a multi-input and single output system and the following type of a fuzzy model for this system:
R^i : **if** x_1 is $A_1^{j_1}$ *and* x_2 is $A_2^{j_2}$ *and* .... *and* x_n is $A_n^{j_n}$ **then** y is B^i,
where R^i is the i-th rule and A_j^i, B^i are fuzzy variables.
In the black box approach of automatic fuzzy model selection from data, we have to build a dynamic model using only empirical input-output data $x_1, x_2, ..., x_n$, y , where x_i inputs and y output data of a dynamic system. Commonly the task of identification is divided in two tasks: structure identification and parameter identification.
In Fuzzy rule induction using GMDH there are realized the following steps:

a. Fuzzification

Fuzzy quantities are expressed by fuzzy numbers or fuzzy sets associated with linguistic labels. The numerical observations of the inputs $\underline{x} = (x_1, x_2, ..., x_n)$ and the output y must be transformed into fuzzy vectors ($\underline{x}^1$, $\underline{x}^2$, ...,$\underline{x}^m$) with $x^j = m_{A^j}(x)$ and y = (y^1, y^2, ...,y^m) with $y^j = m_{B^j}(y)$. The fuzzy membership functions $m_{A^j}(x)$ and $m_{B^j}(y)$ we consider here have a triangular shape.

b. Structure identification: rule generation

Given a class of models (description language) and the data type (fuzzy sets), the task of system identification is to find a model that may be regarded as equivalent to the objective system with respect to input-output data. Such a task of structure identification has to solve two problems: to find out input variables and to find input-output relations.
Self-organizing fuzzy modeling solves both tasks, selecting a finite number of relevant inputs from all possible input candidates and adaptively creating a fuzzy model with an optimal number of fuzzy rules. The rules are written in an IF/THEN-form. According to the number m of output fuzzy variables, m static or dynamic fuzzy models have to be generated (for m=7: y-NB, y-NM, y-NS, y-ZO, y-PS, y-PM, y-PB, where NB-negative big, NM-negative medium, NS-negative small, ZO-zero, PS-positive small, PM-positive medium, PB-positive big).
For self-organizing fuzzy modeling using GMDH algorithms (fig. 2), in the first layer every input represents an input fuzzy set. The number of inputs in the first layer is determined by the total number of fuzzy sets (m) for input variables

(n). Therefore in a static model there are nm neurons where n- number of inputs and m - number of fuzzy variables. If the model is a dynamic one, there are n (L+1) m input neurons, where L is the maximum time lag.

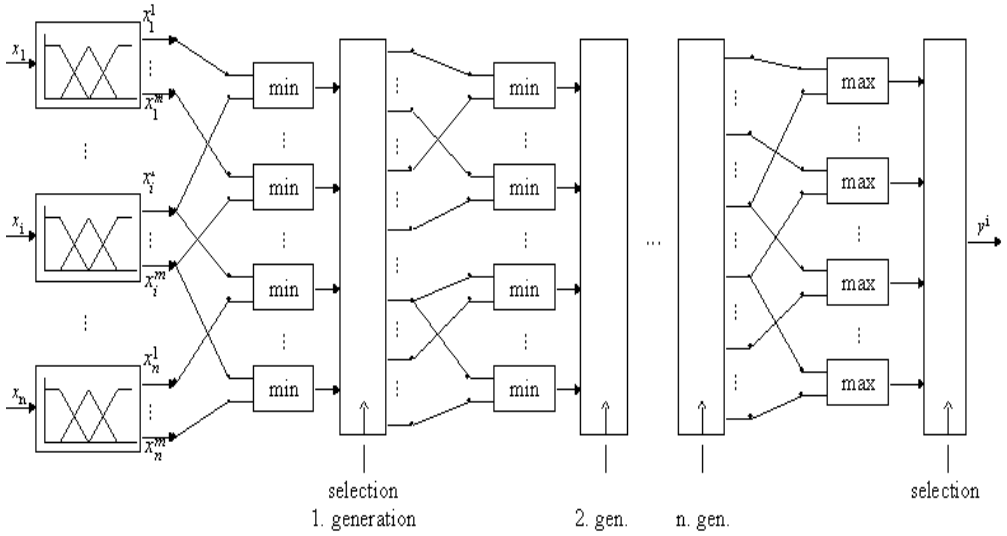

Figure 2: Multi-layer self-organizing fuzzy modeling

Fig. 2 shows a multi-layer architecture, where every neuron has two inputs (x_i^j , x_k^l) and one output (y^r) , which realizes:

IF $x_i^j \wedge x_k^l$ THEN $y^r(i,j,k,l)$.

Using fuzzy logic, those links establish the antecedent relation that is an "AND“ association for each fuzzy set combination. The method for fuzzy inference uses the most general max-min method: $y^r(i,j,k,l) = \min(x_i^j , x_k^l)$.

For the fuzzy output variable y^r (r=1(1)m) in the first layer for all pairs of inputs (i=1(1)n, k≥i, j=1(1)m, l=1(1)m) fuzzy model outputs $y_t^r(i,j,k,l)$ are evaluated for all realizations t = 1(1) N. After the generation of all possible combinations, the F best rules of two fuzzy variables can be selected and used in the following second layer as inputs to generate fuzzy rules of 2,3 or 4 fuzzy variables. The following selection criterion can be used

$$Q_p(i,j,k,l) = \sum_{t=1}^{N} \left| y_t^r(i,j,k,l) - y_t^r \right|^p .$$

Such a procedure can be repeated up to an increasing sum of the F values of criterion of selected fuzzy models. After this, in a second run of self-organization disjunctive combinations of F best models are generated, which can be evaluated by $y_n^r(i,j) = \max(y_n^i, y_n^j)$, where y_n^i, y_n^j - outputs of the n-th generation (last layer), i=1(1) F, j≥i .

IF NM-Dol$_{t-9}$ & PB-FAZ$_{t-5}$ **THEN** NB-BMW$_t$
IF NS-BMW$_{t-3}$ & PM-Ford$_{t-8}$ & ZO-FAZ$_{t-3}$ ∨ ZO-Dol$_{t-4}$ & NS-BMW$_{t-3}$ & PM-Ford$_{t-8}$ & ZO-FAZ$_{t-3}$ **THEN** NM-BMW$_t$
IF ZO-Dol$_{t-1}$ & ZO-Dol$_{t-4}$ & ZO-Dol$_{t-2}$ & PS-DAX$_{t-9}$ & PS-Ford$_{t-5}$ & ZO-DAX$_{t-7}$ & PS-DAX$_{t-4}$ **THEN** NS-BMW$_t$
IF NS-Ford$_{t-5}$ & ZO-Ford$_{t-7}$ **THEN** ZO-BMW$_t$
IF NS-FAZ$_{t-10}$ & PS-DAX$_{t-2}$ **THEN** PS-BMW$_t$
IF NM-FAZ$_{t-4}$ & NS-DAX$_{t-4}$ & ZO-FAZ$_{t-5}$ ∨ ZO-Ford$_{t-5}$ & NM-FAZ$_{t-4}$ & NS-DAX$_{t-4}$ & ZO-FAZ$_{t-5}$ **THEN** PM-BMW$_t$
IF ZO-Dol$_{t-1}$ & ZO-Dol$_{t-4}$ & PB-VW$_{t-3}$ & PB-Ford$_{t-2}$ **THEN** PB-BMW$_t$,

Table IV:. Fuzzy rules for BMW stock rate

c. Defuzzification

The fuzzy output y^r can be transformed back into the original data space by a third run of self-organization using GMDH. As a result, an optimized transformation $y^* = f(y^1, y^2, ...,y^r)$ will be obtained that excludes redundant or unnecessary fuzzy outputs.
Since only relevant fuzzy sets are considered in this way, also information on the optimal number of fuzzy sets is provided implicitly for the given membership function. Using this information for an optimized fuzzification, a complete new run of the rule induction process may result in an increased descriptive and predictive power of the models.

Example : Financial forecasting

Table IV shows a model generated for BMW stock rate on the base of 100 daily close prices of 10 variables of the German stock market.

3.4 SELF-ORGANISATION OF LOGIC BASED RULES

The algorithm of self-organizing fuzzy-rule induction described above can be employed also for generating logic based rules. In this special case, the variables x_i are of Boolean type, for instance, $x_i = 0$ or $x_i = 1$. Instead of the n(L+1)m input neurons used for self-organizing fuzzy-rule induction (fig. 2), there are now only 2 n(L+1) input neurons using both the Boolean variables x_i and their negation NOT x_i. In this way, logical IF-THEN rules can be induced from data such as

IF B_bmw(t-1) & B_dj(t-6) OR B_dj(t-1) & NOT_B_dax(t-2) OR B_dj(t-1) & NOT_B_dax(t-10) OR B_bmw(t-1) & NOT_B_dj(t-3) OR B_bmw(t-1) & B_dj(t-6)
THEN buy_BMW(t),

where buy_BMW is a trading signal (buy) in a trading system, which was generated by means of „KnowledgeMiner“ in the period April 1, 1997 through March 6, 1998 on base of DAX, Dow Jones and BMW stock indexes, and the DOLLAR/DM exchange rate. The Boolean variables are

$$B\_x(t) = \begin{cases} 1 & x(t)>0 \\ 0 & \text{else} \end{cases}, \text{ where } x(t) = \frac{X(t+1)-X(t)}{X(t)}.$$

4. APPLICATION

4.1 APPLICATION FIELDS

The application field is decision support in economics (analysis and prediction of economical systems, market, sales and financial predictions, balance sheet predictions) [Müller (1998), Lemke (1997)] and in ecology (analysis and prediction of ecological processes like air and soil temperature, air and water pollution, growth of wheat, drainage flow, Cl- and NO3-settlement, influence of natural position factors on harvest) [Müller (1996), Wildeshaus (1998)] but also in other fields such as medicine/biology, sociology, engineering, meteorology with only small a priori knowledge about the system.

4.2 SELF-ORGANIZING DATA MINING FOR A PORTFOLIO TRADING SYSTEM

The goal of the trading system is to generate trading signals to provide some decision aid for when to buy or sell a specific asset advantageously. This is usually seen by calculating many trading indicators on historical data. A predictive control solution can be realized if the trading signals are generated also from predictions of the assets of a given portfolio.
The task of self-organizing data mining is to derive a trading signal from data using two kinds of models: one or more prediction models and one or more decision models. In a modeling /prediction module self-organizing data mining is used to extract and synthesize hidden knowledge from a given data set systematically, fast and explicit visible. The con-

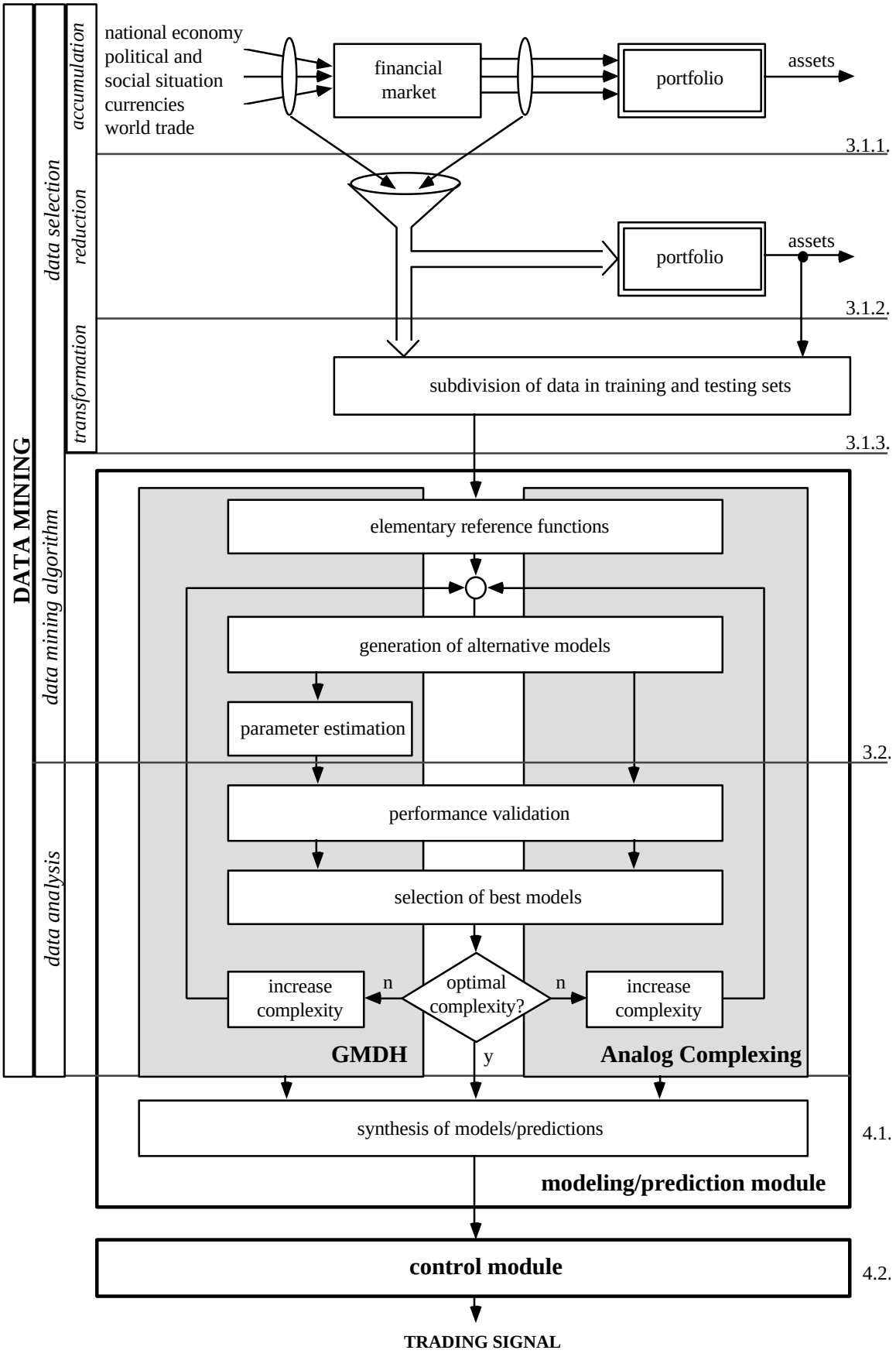

Figure 3: Self-organizing data mining for a trading system [Lemke, 97]

trol module is responsible for signals generation based on a decision model that considers the predictions provided by the modeling module.
In both the modeling and the control module several self-organizing data mining algorithms are differently applicable [Lemke (1997)]. Such a trading system is shown in figure 3. In our performance tests we have tested 2 stocks of the German car industry (BMW, VW) and the US-Dollar/DM exchange rate in the period November 28, 1994 through December 11, 1995 but also others (table V). All trades are computed without commission costs on the corresponding close price one day after a trading signal was generated. No stops or profit targets were used as well as no interest was earned. All data are close prices (more details look at Lemke (1997)). Since our approach prefers a systematical, daily model adaptation due to time variance of financial processes, these tests are true out-of-sample results for the mentioned periods. The total returns of BMW, VW, Dollar and the portfolio are listed in table VI for all 4 strategies: buy&hold, Moving Average Convergence/Divergence (MACD), predictive controlled MACD (MODMACD) and GMDH based.

portfolio	period
BMW, VW, US-Dollar/DM exchange rate	November 28, 1994 - December 11, 1995
S&P500	March 13, 1992 - December 30,1994
DAX, Microsoft, IBM, SAP, Siemens, VEBA, VIAG	September, 6, 1996 -March 3, 1997
30 stocks of DAX and 23 other national and foreign equities	August 30, 1996 - March 6, 1998
DJIA, DAX, Dollar	July 7, 1997 - April 1, 1998
Intel Corporation, Novell, Sun Microsystems, Apple Computer	April 22, 1996 - April 17, 1998

Table V:. Tested stocks

strategy	BMW		VW		Dollar		Portfolio
	time[%]	return[%]	time[%]	return[%]	time [%]	return[%]	return [%]
buy&hold	100,00	0,13	100,00	7,89	100,00	-8,50	-0,48
MACD	60,15	0,73	50,92	10,70	37,27	-7,34	4,09
MODMACD	41,70	14,29	53,51	29,20	57,56	-4,50	38,99
GMDH based	52,40	22,55	66,05	41,27	58,67	8,04	71,86

Table VI:. Total returns (November 28, 1994 - December 11, 1995) for different trading strategies (time = time in market; return = total return)

4.3 SOLVENCY CHECKING

The goal is to find a model that classifies based on a company's recent balance sheet whether or not it is qualified in the bank's sense to get some credit. Exactly it is the bank's decision policy that should be learned from the given data as accurate as possible. It was not guaranteed, however, that this policy would have been consistent, i.e. that the data samples contain not any false decision. This makes modeling and interpretation of the results much more difficult since it cannot avoided that a model may also reflect false decision.
Basis for the examination and automatic model synthesis were sets of 19 characteristics of 81 companies which have been served a banking establishment to decide a company's solvency. 10 decisions have been chosen from the bank to serve for results checking while the other 71 decisions (35 positive and 36 negative) were used as learning data set for modeling. The checking data set contains 5 positive, 3 negative and 2 undecided.
There are several methodologies for obtaining the required models using self-organizing data mining, but in distinction to neural networks each of them also provides an explanation component.

Linear and nonlinear GMDH models

The decision variable was described by linear or nonlinear models correspondingly from the 19 characteristics x_i. As significant variables $x_7,x_9,x_{11},x_{12},x_{13},x_{15},x_{18}$ were selected for the linear and $x_3,x_4,x_5,x_6,x_8,x_{11},x_{13},x_{14},x_{18}$ for the nonlinear model. Table VII lists the classification results for nonlinear model.

Fuzyy-GMDH

Fuzzy-GMDH has created the following two rules:
IF NOT NS_x10 & NOT_NS_x6 & NOT_ZO_x11 & ZO_x13 & NOT_NB_x6 & ZO_x19 & NOT_PB_x13
THEN positive decision
IF NOT_ZO_x19 OR NOT-ZO_x13 OR NOT_PB_x7 & NOT_PS_x6 & NOT_PB_x6 OR ZO_x11 OR NOT_PB_x10 & NOT_PS_x6 & NOT_PB_x6
THEN negative decision.
This is a more natural description of the problem using another set of variables in the premise parts, but only x_{10} and x_{19} are new contributors here compared with the GMDH solution. The table VII shows the classification power here.

Analog Complexing

For the test case t1, for example, AC has selected these instances as similar cases compared with t1: p3, n26, p7, p6, p9 with high degree of similarity. Using a majority decision, this model suggests a comfortable positive vote. The results are included in table VII.

case	target	GMDH (nonlinear	Fuzzy - GMDH	Analog Complexing
t1	n/p	n	n	p
t2	n/p	n	n	p
t3	p	p	p	p
t4	n	n	n	p
t5	p	p	p	p
t6	n	n	n	n
t7	n	n	n	n
t8	p	p	n	p
t9	p	p	p	p
t10	n	n	n	n

Table VII:. Classification results using self-organizing data mining

5. CONCLUSIONS

- Obviously, inductive methods cannot substitude the necessary analysis of causes of events by means of theoretical systems analysis, but clever applications of these tools may reveal carefully guarded secrets from nature. A pragmatic solution to the model building problem is an union of the deductive and inductive methodologies. One development direction that take up the practical demands represents selecting models from data which is realizable by means of neural networks as well as by self-organizing modeling like GMDH algorithms.
- Models obtained by self-organizing modeling are non-physical. A non-physical model is a simplified physical model, which can be obtained by exclusion of some members from equation of physical model. In conditions of noised and short data sample, non-physical model, which gives the most accurate approximation and process forecasting, can be obtained by GMDH algorithms only. These algorithms realize sorting-out procedures by external accuracy type criteria to find non-physical optimal model.
- Objects with fuzzy characteristics can be described by
  - clusters or patterns using Objective Cluster Analysis or Analog Complexing. Clusters/patterns are defined by selection type GMDH sorting-out algorithms.
  - systems of fuzzy rules.
- To estimate the vagueness of prediction and to increase robustness and reliability of prediction different predictions, automatically generated by means of these different technologies must be synthesized. The reason for synthesizing models/predictions is that each model or pattern reflect only a specific behavior of reality. By combining several models it is more likely to reflect reality in a more complete and robust fashion.
- "KnowledgeMiner" is a powerful easy -to- use modeling and prediction tool which
  - was designed to support the knowledge extraction process from data on a highly automated level;
  - works on advanced self-organizing modeling technologies ;

- requires only minimal, uncertain a priori information about the system;
- deals with data like in spreadsheets;
- creates linear or nonlinear time series models, multi-input/single-output models and systems of equations or fuzzy rules for multi-input/multi-output systems;
- creates a best and autonomous system of equations (network of GMDH-type Neural networks) which is ready for status-quo predictions of the complete system by default and which is available analytically and graphically (system graph) for results interpretation;
- creates non-parametric prediction models for fuzzy objects by Analog Complexing, an advanced pattern search technology for evolutionary processes;
- generates analytical models as a explanation component;
- stores created models in model base, which are applicable immediately to sets of new data (prediction, classification, diagnosis).

- Self-organizing data mining algorithms for a portfolio trading system were presented realizing a predictive control solution. To get predictions for financial markets appropriated for decision making, there has been realized a moving modeling by using:
  - GMDH-type Neural Networks to create automatically optimal complex, parametric regression models which are analytically available by default. It was shown that GMDH is also suitable to solve several subtasks for data reduction, synthesis and rule induction in a fast and systematical way.
  - Analog Complexing as a method to select similar market situations out of a given data set of representative variables and a
  - Synthesis of different models to reflect the vagueness of future more appropriately.
- A second task of the trading system was to transform the obtained predictions into trading signals. There have been tested two options:
  - a modified MACD indicator and
  - a synthesis of several types of predictive information using GMDH.

  Initial performance results have shown that the realized predictive control solution seems to be able to outperform a buy-and-hold strategy as well as a MACD-based trading system in a long run and for various assets (table VI).
- Self-organizing data mining can provide useful information for solvency checking since each applied method has, in contrast to neural Networks, some explanatory power that allows users analyzing why a model's decision is as it is. This is a key factor that was formulated explicitly by the bank as absolutely necessary feature of any solution. The ability for using different description languages with the spectrum of modeling methods, and thus capturing different behavior, is important also when trying to reflect the problem's complexity. Having a fuzzy decision variable that indicates clear and uncertain decisions could be helpful here also. Self-organizing data mining might be a valuable, objective decision aid for solvency checking when considering that a solution for this problem is not only a matter of numbers allowing complete automation.

REFERENCES

Bigus, Joseph P., 1996, "Data Mining with neural networks", McGraw Hill, New York.

Fayyad, Usama M. et al, 1996, "From Data Mining to Knowledge Discovery: An Overview", In : „Fayyad, Usama M. et al, 1996 "Advances in Knowledge Discovery and Data Mining", AAAI Press/The MIT Press. Menlo Park, California“, pp. 1-36.

Kingdon, Jason, 1997, "Intelligent Systems and Financial Forecasting", Springer, London, Berlin, ..

Lemke, Frank; Müller, Johann-Adolf, 1997, "Self-Organizing Data Mining for a Portfolio Trading System". Journal of Comp. Intelligence, 5, No.3, pp. 212-26.

Madala, Hema R.; Ivakhnenko, Aleksej G., 1994, "Inductive Learning Algorithms for Complex Systems Modelling", CRC Press Inc., Boca Raton, Ann. Arbor, London, Tokyo.

Müller, Johann-Adolf, 1996, "Analysis and prediction of ecological systems", SAMS, vol.25, pp.209-243.

Müller, Johann-Adolf, 1998, "Automatic Model Generation" , SAMS, vol.31, No. 1-2, pp. 1-32.

Wildeshaus, Thilo, 1998, "GMDH-Algorithmen und deren Anwendung zur Erstellung von Vorhersagemodellen für die Gewässergüte der Spree". Diplomarbeit an der Rheinisch-Westfälischen Technischen Hochschule Aachen.