

**J.-A-Müller**

# **Data Mining und automatische Modellgenerierung**

## **1. Data Mining**

Die moderne Informationstechnologie liefert uns eine Informationsflut, deren vollständige Überwachung nahezu ausgeschlossen ist. Allein die Bereitstellung von Daten führt noch nicht zu einer Verbesserung in der Qualität der Entscheidungsfindung. Erforderlich werden entsprechende Tools, um automatisch aus diesen Daten benötigtes Wissen zu extrahieren. Wissensextraktion aus den Daten (knowledge discovery from data-KDD) stellt einen umfangreichen Prozeß dar von der Aufgabenformulierung, Datenselektion, -vorverarbeitung und- reduktion über die Auswahl und Anwendung geeigneter Data Mining Algorithmen bis hin zur Analyse des extrahierten Wissens und seiner Interpretation und Bewertung aus der Sicht der Entscheidungsaufgabe. Data Mining selbst beinhaltet den Versuch, mittels verschiedener Verfahren zum Visualisieren und Extrahieren von Wissen aus dieser sehr großen Datenbasis bisher unbekannte Zusammenhänge, Muster, Trends u.a. möglichst automatisch zu ermitteln. Dazu gehören neben den bewährten Methoden der mathematischen Statistik auch solche, die es erlauben, aus den Daten für die Entscheidungsfindung benötigte mathematische Modelle zu generieren. Deren weitere Entwicklung und Anwendung im Rahmen des KDD hängt jedoch sehr stark davon ab, inwieweit es gelingt, im Data Mining Prozeß die Interaktion mit dem Anwender zu reduzieren. Dieser ist in der Regel ein nicht programmierender und nicht modellierender Nutzer, der sich nur für die eigentliche Aufgabenlösung interessiert und kaum Spezialkenntnisse für eine dialogorientierte Modellerstellung mitbringt. Da der Nutzer darüberhinaus in der Regel aber auch nicht die notwendige Zeit für den Rechnerdialog aufbringen kann, vielfach massenhaft Modellvarianten erstellt werden müssen und generell wenig Zeit für komplexe Entscheidungen bleibt, wird eine weitgehende automatische Modellgenerierung im Rahmen des Data Mining notwendig.

## **2. Automatische Modellgenerierung**

Im Rahmen einer automatischen (selbstorganisierenden) Modellgenerierung werden u.a. folgende mathematische Modellstrukturen generiert:

### *a. Lineare/nichtlineare Regression*

In der mathematischen Statistik wurde ein umfangreicher Methodenapparat entwickelt, der zur Lösung dieser Aufgaben geeignet ist. Bei seiner Anwendung entsteht eine Vielzahl methodologischer und erkenntnistheoretischer Probleme im Zusammenhang mit der ungenügenden A-priori-Information, der Überprüfbarkeit der diesen Methoden zugr<sup>u</sup>liegenden Hypothesen, der unzureichenden mathematisch-statistischen Kenntnisse der Anwender u.a.. Weit verbreitet sind zur Zeit Bemühungen, im Data Mining die Technologie Neuronaler Netze anzuwenden [Bigus,96], obwohl deren Anwendung im Zusammenhang mit der verteilten Modellierung, der subjektiven Wahl der Netzwerktopologie, Integration der vorhandenen A-priori-Information u.a. [Müller.97] problematisch ist. Für die Aufgaben der Modellierung werden verschiedene Netzwerktopologien angeboten und genutzt, jedoch nicht die eigentlich für die mathematische Modellierung vorgesehenen Neuronalen Netze vom GMDH-Typ (Abschn.3.1), die auch in der Literatur weitgehend unberücksichtigt bleiben.

*b. Selbstorganisierende Generierung von Prozeßmodellen* Die Anwendung genetischer Algorithmen ist zwar sehr rechenintensiv, bietet aber eine sinnvolle Möglichkeit zur selbstorganisierenden Generierung nicht nur Neuronaler Netze (auf diese Weise wird eine Topologie-Optimierung möglich) sondern strukturierter Prozeßmodelle überhaupt. Generiert werden strukturierte Prozeßmodelle oder algebraische Modelle in Baumstruktur. Dieser Ansatz ermöglicht in gleicher Weise sowohl prozeßbezogenes Bedienerwissen (Vorgabe des Inhalts des Modellbaukastens) als auch konkrete Prozeßdaten (Parameterbestimmung und

Bewertung der Modellgüte) zur Modellbildung heranzuziehen [Marenbach, 95]. Die für die Anwendung genetischer Algorithmen erforderliche genetische Darstellung eines Modellindividuums erfolgt durch eine Kette von Genen, wobei jeder Modellblock (Übertragungsglied) einem Gen entspricht. Das einzelne Gen enthält neben der Information darüber, welches Übertragungsglied es repräsentiert, die Anzahl der Eingänge und numerischen Werte der im Modellblock enthaltenen Parameter.

### *c. Nichtparametrische Modelle*

Nichtparametrische Modelle in Form von einem oder mehreren Pattern (zum gegenwärtigen Entwicklungsabschnitt ähnliche Entwicklungsabschnitte der Vergangenheit) werden aus den Daten selektiert. Eine derartige Herangehensweise (Analogiemethode) wird in der Meteorologie seit langem erfolgreich angewendet, da sie hier auf einer bewährten physikalischen Basis beruht ("gleiche Ausgangswetterlage - gleiche Weiterentwicklung") (Abschn.3.2).

Als Modell dient ein Regelsystem der klassischen Logik (meist in disjunktiver Normalform) oder der Fuzzy-Logik. GMDH-Algorithmen ermöglichen eine Selbstorganisation von Fuzzy-Modellen (Abschn.3.3), wobei zusätzlich wie im ersten Fall genetische Algorithmen Anwendung finden können.

## **3. "KnowledgeMiner" - Tool zur automatischen Modellgenerierung [Lemke, 97]**

Der Anwendungsbereich von "KnowledgeMiner" (<http://www.scriptsoftware.com/km>) ist die Modellgenerierung für komplexe Systeme auf der Grundlage von kurzen und verrauschten Datensätzen. Erfolgreiche Anwendungen der GMDH-Algorithmen liegen insbesondere in den Bereichen vor, in denen die theoretische Systemanalyse auf Grund der Kompliziertheit der Untersuchungsobjekte, des Entwicklungsstandes der einzelwissenschaftlichen Theorie aber auch der erforderlichen Zeit nur beschränkt Anwendung finden kann. Das sind neben Anwendungen zur Modellierung volkswirtschaftlicher und betriebswirtschaftlicher Systeme insbesondere Ergebnisse bei der Analyse und Vorhersage ökologischer Prozesse, wie z.B. Luft- und Bodentemperatur, Luft- und Wasserverschmutzung, Drainageabfluß sowie CL- und N03- Austrag [Müller, 96].

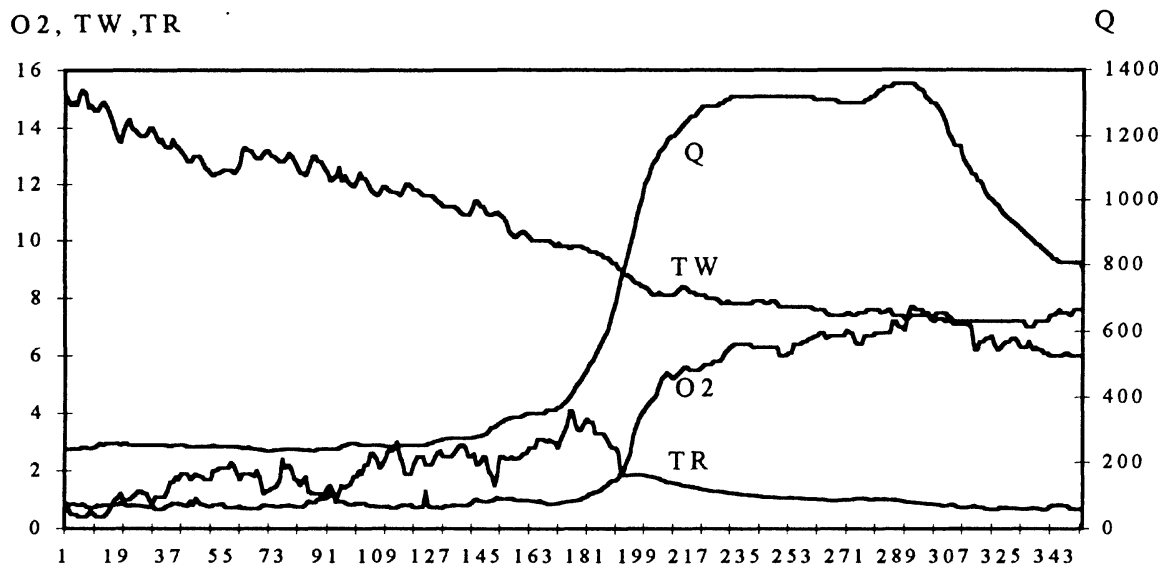
### **3.1 Parametrische Modelle**

Automatisch generiert werden Zeitreihenmodelle sowie Multi-Input - Multi-Output - Modelle (Systeme interdependenter Gleichungen) mit gegenwärtig maximal 500 Inputvariablen. Für Multi-Input - Multi-Output - Modelle können Testverfahren zum Einsatz kommen, die die mathematisch widerspruchsfreie (zyklenfreie) Generierung von interdependenten Gleichungssystemen gewährleisten und dadurch eine schrittweise Vorhersage ermöglichen bei minimaler A-priori-Information über das zu modellierende Objekt. Diese A-priori-Information umfaßt

- die Auswahl geeignet scheinender beobachteter, abgeleiteter oder synthetisierter Variablen, die zur Beschreibung des Objektes als Eingangs- und/ oder Ausgangsgrößen relevant sein könnten,
- die allgemeine Spezifizierung des Modells als linear oder nichtlinear sowie seiner Dynamik. Neben der Synthese eines optimal komplexen Netzwerkmodells findet in "KnowledgeMiner" zusätzlich eine Optimierung der Struktur der einzelnen Transferfunktionen (Neuronen) statt. Für jedes Neuron wird aus einer Klasse allgemeiner Transferfunktionen entsprechend dem PESS-Kriterium die optimale Transferfunktion ausgewählt. Das generierte Netzwerk stellt eine Komposition verschiedener, a priori unbestimmter partieller Modelle dar. Dieser Algorithmus ermöglicht, lineare oder nichtlineare Modelle optimaler Kompliziertheit in Abhängigkeit von der Objektstruktur und einer sinnvollen Reduzierung der Komplexität bezüglich der vorhandenen Rauschdispersion zu generieren. Aufgrund dieser minimalen Erfordernis unsicherer A-priori-Information, begründet in der induktiven Herangehensweise an die Modellierung, ist es möglich, auf einfache, schnelle, konsistente und objektive Weise Analyse-,

Klassifikations- und Vorhersageaufgaben komplexer ökonomischer, ökologischer, sozialer, technischer u.a. Systeme zu lösen. Das durch die erhaltene Netzwerkfunktion repräsentierte Modell steht unmittelbar nach der automatischen Modellgenerierung explizit zur Verfügung und kann somit außer zur Erstellung von Szenarien (Vorhersage von Kennzahlen oder Kennzahlensystemen) auch zur Interpretation und Analysetätigkeit im Rahmen der Modellerstellung herangezogen werden. Eine transparente Dateneingabe und Datenhaltung, wie sie von Tabellenkalkulationsprogrammen her bekannt sind, eine Modellbank sowie eine grafische und analytische Ausgabe aller generierten Modelle erlauben ein komfortables und flexibles Arbeiten mit den erhaltenen Informationen. **Beispiel: Wassergüte Elbe**

Gegeben sind die durch den Betrieb einer automatischen Meßstation am Gütepegel Magdeburg-Strombrücke in dreistündigem Abstand gemessenen Primärdaten folgender Wasserkenngrößen für den Zeitraum vom 1.10.1974 bis 31.12.1974 : O<sub>2</sub> - Sauerstoffgehalt, TW - Wassertemperatur, Q - Abfluß, TR - Trübe. Im Bild dargestellt sind diese Kenngrößen für den Zeitraum vom 7.10.-13.11.1974.



Mit Hilfe von "KnowledgeMiner" ermittelt wurden Modelle dieser Größen in Form von linearen und nichtlinearen Differenzgleichungssystemen sowie entsprechende Vorhersagen. Für den Zeitraum vom 1.10.-11.11. (330 Realisierungen) wurde u.a. folgendes lineare Modell erhalten (maximale Verzögerung 40 und damit 163 Input-Variable, 4 Output-Variable):

$$O_2(t) = 0.82 + 0.96 O_2(t-1) - 0.084 O_2(t-3) - 0.033 O_2(t-10) + 0.031 O_2(t-22) - 0.048 TW(t-6) + 4 \cdot 10^{-4} Q(t-5) - 0.076 TR(t-4)$$

$$TW(t) = 0.06 + 0.96 TW(t-1) + 0.109 TW(t-7) - 0.174 TW(t-10) + 0.034 TW(t-23) - 0.039 TR(t) \quad Q(t) = 14.16 + 1.18 Q(t-1) - 0.188 Q(t-6) - 1.179 TW(t-14) + 3.626 TR(t-35)$$

$$TR(t) = 0.0214 + 0.978 TR(t-1).$$

Für drei Beobachtungszeiträume (a: 1.-13.10., b: 29.10.-11.11., c: 1.10.-11.11.) mit jeweils 100 (a,b) bzw. 330 Realisierungen (c) - wurden mit Hilfe entsprechender Modelle Langfristvorhersagen für 11 bzw. 22 Realisierungen generiert. Deren Fehler (MAD [%]) enthält Tafel 1

	a/11	a/22	b/11	b/22	C/11	c/22
02	19.85	25.37	4.47	6.4	1.50	1.14
TW	13.34	12.04	1.13	2.66	1.53	3.98
O	2.88	2.31	3.00	8.01	2.72	6.26
TR	7,75	21,26	6,04	7,39	5,49	7,52

Tafel 1: Fehler (MAD [%]) der Langfristvorhersagen

### 3.2 Nichtparametrische Modelle

In "KnowledgeMiner" ist darüberhinaus die Analogiemethode enthalten, die selbständig oder als Ergänzung für What-if-Vorhersagen genutzt werden kann. Die Analogiemethode geht von folgenden Annahmen aus:

- • das betrachtete System läßt sich durch einen mehrdimensionalen Prozeß beschreiben,
- • der mehrdimensionale Prozeß ist ausreichend repräsentativ, d.h. die für das Verhalten des Systems wesentlichen Systemgrößen sind erfaßt.

Unter diesen Annahmen kann man davon ausgehen, daß sich Entwicklungsabschnitte der Vergangenheit wiederholen können. Gelingt es, einen solchen zum gegenwärtigen Entwicklungsabschnitt analogen Abschnitt zu ermitteln, so läßt sich die Vorhersage aus der in der Vergangenheit bereits bekannten Weiterentwicklung des ermittelten Analogs (oder mehrerer ermittelter Analoge) bestimmen.

Die im "KnowledgeMiner" realisierte Herangehensweise ist auch für evolutionäre Prozesse anwendbar, wobei für evolutionäre Prozesse anstelle einer Trendelimination davon ausgegangen wird, daß die Weiterentwicklung durch eine geeignet gewählte Transformation erfaßt werden kann. Zur Ermittlung der für die Vorhersage geeigneten Pattern entsteht nunmehr das folgende Auswahlproblem:

Ausgehend von dem Ausgangspattern sind ein oder mehrere diesem Pattern ähnliche Pattern zu suchen, mit deren Hilfe die Entwicklung im Vorhersagezeitraum ermittelt werden kann. Dabei sind die Systemgrößen, die Breite der Pattern und die Anzahl der zur Vorhersage zu verwendenden Analoge  $F$  zu bestimmen, die für die Vorhersage die besten Ergebnisse bringen. Schließlich ist durch eine geeignete Kombination der ausgewählten Vorhersagen die endgültige Vorhersage zu generieren. Es können Auswahlalgorithmen unter Nutzung von Bilanzkriterien (bei Verwendung verschiedener Diskretisierungen) oder der Crossvalidation Anwendung finden. *Wassergüte Elbe:* Für die Beobachtungszeiträume  $a, b$  und  $c$  wurden Pattern der Breite 5 bis 14 ausgewählt und zu Langfristvorhersagen mit 11 bzw. 22 Realisierungen kombiniert. Tafel 2 zeigt die erreichten Vorhersagefehler (MAD [%]).

	a/11	a/22	b/11	b/22	C/11	c/22	bs/22	cs/22
02	16.61	10.55	5.88	6.79	3.58	3.77	6.60	2.33
TW	3.56	2.83	1.91	4.14	1.16	2.96	3.40	3.47
O	1.62	1.05	3.14	3.20	3.94	5.56	5.60	0.97
TR	10.97	7.62	6.17	14.20	3.78	5.89	6.38	6.20

Tafel 2: Fehler (MAD [%]) der mit Analogiemethode erreichten Langfristvorhersagen

Aus Tafel 2 ist den Spalten c/11 bzw. c/22 zu entnehmen, wie sich eine Erweiterung des Beobachtungszeitraums auf die Vorhersagegenauigkeit auswirken kann.

### 3.3 Selbstorganisierende Fuzzy-Modellierung

In der Weiterentwicklung von "KnowledgeMiner" ist eine Erweiterung um eine regelbasierte Modellierung in Form von Fuzzy-Modellen vorgesehen, um dem "principle of incompatibility" von Zadeh ("In dem Grad, wie die Komplexität eines Systems steigt, nimmt unsere Fähigkeit ab, präzise und zugleich bedeutsame Aussagen zu machen. Oberhalb einer gewissen Komplexitätsschwelle werden Präzision und Relevanz fast

gegensätzliche Charakteristiken" [Kosko, 94]) zu genügen. Fuzzy-Modelle erlauben eine adäquate Beschreibung der Unbestimmtheit der Untersuchungsobjekte. Für ein System mit den linguistischen Variablen: Eingangsgrößen  $X_1, \dots, X_n$  und Ausgangsgröße  $y$  wird eine Beschreibung durch das folgende Regelsystem gesucht:

**if**  $x_1$  is  $A^1$  and  $x_2$  is  $A^2$  and ... and  $x_n$  is  $A^n$  **then**  $y$  is  $B^1$  wobei  $A^j, B^1$  linguistische Terme darstellen. Derartige Regelsysteme können im Ergebnis der theoretischen Systemanalyse ermittelt werden oder aber im Data Mining aus den empirischen Beobachtungen  $(X_t, y)$  selbst. Das beinhaltet neben der Fuzzification selbst die Aufgabe der Struktur- und Parameteridentifikation. *I. Fuzzification*

Jeder Komponente des Inputvektors  $x = (x_1, x_2, \dots, x_n)$  und des Outputs  $y$  des Systems wird ein  $m$ -Tupel ( $m$ -Anzahl linguistischer Terme) von Zugehörigkeitsgraden zugeordnet. Damit ergibt sich ein Fuzzy-Inputvektor  $(x^1, x^2, \dots, x^n)$  mit  $x^i = \mu_{A^i}(x)$  und ein Fuzzy-Outputvektor  $y = (y^1, y^2, \dots, y^m)$  mit  $y^i = \mu_{B^i}(y)$ . Die Zugehörigkeitsfunktionen  $\mu_{A^i}, \mu_{B^i}$  können im einfachsten

Fall vom Lambda-Typ gewählt werden. *II. Strukturidentifikation : selbstorganisierende Regelgenerierung*  
Mit Hilfe der GMDH-Algorithmen wird für jede Fuzzy-Outputkomponente  $y^i$  ( $i=1 \dots m$ ) eine Regel erstellt, in die theoretisch alle Komponenten des Fuzzy-Inputvektors (statisch) oder alle Komponenten mit einer Zeitverzögerung  $t=l(1)L$  (dynamisch) eingehen können. Dabei hat das sich ergebende Netzwerk  $n(L+1)m$  Inputneuronen und  $m$  Outputneuronen. Jedes Neuron hat zwei Inputs und einen Output. Somit gilt in der ersten Generation zur Generierung z.B. eines statischen Modells:

**if**  $x_1$  and  $x_2$  **then**  $y^i$ , wobei  $(x_1, x_2)$  - Inputs und  $y^i$  - Output.

Dabei gilt für den Output der Neuronen:

$y^i(j,k,l) = \min(x_1, x_2)$ , wobei paarweise alle möglichen Modelle generiert werden. Die besten Modelle werden selektiert und deren Outputs als Inputs der zweiten Generation verwendet. Bei geeignetem Selektionskriterium ergeben sich Fuzzy-Modelle optimaler Kompliziertheit, die in der letzten Generation noch disjunktiv verknüpft werden können. *III. Parameteridentifikation*

Die in der Fuzzification verwendeten Zugehörigkeitsfunktionen enthalten Parameter, die entweder durch Experten vorgegeben sind, automatisch gewählt werden oder aber mit Hilfe genetischer Algorithmen bestimmt werden können. Im letzteren Fall erfolgt eine Aufteilung der Beobachtungen in Lern- und Prüffolge und damit die Möglichkeit, externe Information zur Auswahl eines Modells optimaler Kompliziertheit zu nutzen. *IV. Defuzzifikation*

Ziel des Data Mining im eigentlichen Sinne ist nicht die Modellierung sondern die Entscheidungsfindung. Insofern ist es sinnvoll, die auf diese Weise generierten Fuzzy-Modelle direkt zur Erzeugung von Fuzzy-Outputvektoren (für gegebene Fuzzy-Inputvektoren) zu verwenden und auf deren Grundlage in geeigneter Weise eine Entscheidung herbeizuführen (siehe Abschn.4). Werden scharfe Ausgangsgrößen erforderlich, können die bekannten Methoden zur Defuzzifikation angewendet werden.

**Wassergüte Elbe:** Mit einem im Leistungsumfang stark beschränkten Prototyp für die selbstorganisierende Fuzzy-Modellierung wurden für verschiedene Zeitabschnitte Fuzzy-Modelle erstellt und mit deren Hilfe Vorhersagen ermittelt. Zur Generierung verwendet wurden Differenzen der Inputs ( $\Delta X_t = X_t - X_{t-1}$ ) und Outputs ( $\Delta y_t = y_t - y_{t-1}$ ). Unter anderem wurde für 5 linguistische Terme das in Tafel 3 dargestellte Modell für  $0 \leq t \leq 11$  im Zeitraum 29.10.-11.11. (100 Realisierungen) erhalten.

NB-AÖ2  $t=ZO-ATWt.i$  &  $ZO-ATWt.2$  & NB-  $ATWi.g$  & NB- $ATWt.i5$  NS-AÖ2  $i=NS-ATWt.6$  & NS-  
AQi.15 &  $ZO-ATR^U$  NS- $ATWi-15$  & NS- $AQi.g$  &  $ZO-AQi.io$  &  $ZO-ATRi.io$   $ZO-A02t=ZO-TRt.i3$  U  
 $ZO-AQi.i4$  &  $ZO-AQi.u$

PS-A02t=NS- $ATWt.i3$  &  $ZO-ATRt.i3$  &  $ZO-ATRt.n$  PB-AÖ21 = NB- $ATWf.5$  & NS- $ATRt.H$  & PB- $ATWi.i4$

NB-negative big, NM-negative medium, PS- negative small, ZO-zero, PS-positive small, PM-positive medium, PB-positive big.

Tafel 3: Fuzzy-Modell

	5 Terme		7 Terme		5 Terme		7 Terme	
	a/11	a/22	a/11	a/22	b/11	b/22	b/11	b/22
Oz	1.94	15.35	3.53	8.27	6.86	20.58	9.29	28.10
TW	1.25	5.81	1.08	4.83	2.11	1.48	1.21	2.29
O	2.01	2.95	2.31	2.90	1.58	2.32	2.19	0.92
TR	5.92	15.39	3.84	12.88	8.23	42.12	5.11	49.56

Tafel 4: Vorhersagefehler (MAD[%]) der Fuzzy-Modelle

Tafel 4 enthält die Fehler der mit den ermittelten Fuzzy-Modellen berechneten Vorhersagen (MAD [%]) für 11 bzw. 22 Realisierungen für die bereits erwähnten Beobachtungsintervalle a bzw. b. Die Fuzzy-Outputvariablen wurden mit Hilfe der Schwerpunktmethod in eine scharfe Größe überführt wurden.

#### 4. Schlußfolgerung

Data Mining hat nicht die Aufgabe, Modelle an sich möglichst automatisch aus den Daten zu generieren sondern Entscheidungen zu unterstützen. Deshalb sind die auf die dargestellte Weise generierten Modelle nicht das Endergebnis sondern einzubetten in entsprechende Anwendungslösungen. Vorgestellt wurden verschiedene Verfahren zur automatischen Generierung von Modellen bzw. Vorhersagen aus den Daten. Bezüglich Rechenzeit sind dabei die parametrischen Modelle im Verhältnis zur erzielten Genauigkeit der Langfristvorhersagen zu aufwendig. Eine umfangreiche Aufgabenklasse stellt zweifellos die prediktive Steuerung dar, d.h. auf Grundlage der generierten Modelle ist Information über zukünftige Verhaltensvarianten zu erstellen (Vorhersagemodul), um damit einen rechtzeitigen Steuerungseinfluß vorzunehmen (Steuermodul). Derartige Ansätze sind nicht für den allgemeinen Fall zu entwickeln sondern erfordern eine aufgabenspezifische Lösung. In [Lemke/Müller, 97] ist eine entsprechende Referenzlösung für ein Portfolio-Tradingssystem in der Finanzanalyse vorgestellt. Vorteil selbstorganisierender Data Mining Algorithmen, wie sie im vorliegenden Beitrag vorgestellt wurden, ist, daß sie mit einer geringen Stichprobe und wenig Zeit erlauben, alternative Vorhersagen (nichtparametrische bzw. parametrische, lineare und nichtlineare Modelle mit verschiedener Zeitverzögerung, Fuzzy-Vorhersagen u.a) zu generieren. Deren Synthese ergibt auf Grund des Mengenprinzips eine adäquatere Beschreibung komplizierter Systeme und in der Regel reduzierte Vorhersagefehler (siehe Tafel 2 bs , es: Synthese von Vorhersagen, die mit GMDH bzw. Analogiemethode für die Zeiträume b bzw. c erhalten wurden). Gleichzeitig wird auf diese Weise die Erfassung der Unbestimmtheit möglich. Diese sowie eine geeignete Kombination der Vorhersagen eventuell auf der Grundlage entsprechender Fuzzy-Modelle ergeben entsprechende Entscheidungsregeln im Steuermodul.

#### Literatur

Bigus, J.P.: Data Mining with Neural Networks. MC Graw Hill 1996

Kosko, B.: Fuzzy Thinking. Flamingo. An Imprint of Harper Collins Publishers 1994.

Lemke, F.: Knowledge Extraction from Data Using SelfOrganizing Modeling Technologies. eSEAM'97 Conference. MacSciTech organization 1997.

Lemke, F., Müller, J.-A.: Self-Organizing Data Mining for a Portfolio Trading System. Journal of Computational Intelligence in Finance. vol. 5 (1997) No. 3, 12-26.

Marenbach, P., K.D. Bettenhausen, B. Cuno: Selbstorganisierende Generierung strukturierter Prozeßmodelle, at 43 (1995) 6, 277-288.

Miller, G.F., Todd, P.M.(1989) Designing Neural Networks using Genetic Algorithms. In: Proc. of the 3. Intern. Conf. on Genetic Algorithms. Morgan Kaufmann Publ. 1989. pp.379-384.

Müller, J.-A.: Analysis and prediction of ecological Systems. SAMS vol. 25 ,1996, pp.209-243

Müller, J.-A., Ivachnenko, A.G., Lemke, F: GMDH Algorithms for complex Systems modelling . Mathematical Modelling of Systems, vol.2, 1997

**Autorenangabe:**

Hochschule für Technik und Wirtschaft Dresden Fachbereich

Informatik/Mathematik Prof. Dr.rer.oec.habil. J.-A. Müller

F.-List-Platz 1 Dresden 01069 Tel. 0351-462 3322 fax: 0351

462 3671 email: muelleij@informatik.htw-dresden. de