

ANALYSE UND VORHERSAGE MITTELS STATISTISCHER LERNENDER NETZE

Anwendersymposium zur industriellen Anwendung der Neuro-Fuzzy-Technologien. Wittenberg 1995.

Frank Lemke
DeltaDesign Software
Bergener Str. 1, Berlin 10439
Tel. (030) 444 3585 Fax (030) 279 2984

Johann -Adolf Müller
Hochschule für Technik und Wirtschaft
F.- List - Platz 1, Dresden 01069
Tel. (0351) 462 3322, Fax (0351) 462 3671

1. Rechnergestützte Modellauswahl

Eine praktikable Modellierung muß den interaktiven Modellentwurf auf der Grundlage der theoretischen bzw. experimentellen Systemanalyse, der durch reichhaltige Software gegenwärtig bereits unterstützt wird, mit Elementen einer rechnerunterstützten (automatischen) Modellwahl verbinden. Als Gründe dafür kann man zusammenfassend aufzählen:

1. Eine sachkundige Anwendung der Software stellt oftmals an die Nutzer zu hohe Anforderungen;
2. Viele Anwender interessieren sich nur für die Aufgabenlösung, sie bringen kaum Sachkenntnis über mathematisch-statistische Verfahren mit und schon gar nicht die für den Rechnerdialog erforderliche Zeit;
3. In vielen praktischen Fällen sind massenhafte Vorhersagen bzw. Analysen erforderlich, die erforderliche Zeit für komplexe Untersuchungen fehlt.

Versuche, wissensbasierte Methoden der Künstlichen Intelligenz zur Auswahlberatung, Nutzerführung und Interpretation der erhaltenen Ergebnisse einzusetzen, brachten vielfach nicht die erwünschten Ergebnisse. Ursache dafür ist einerseits, daß diese Verfahren auf einer Extraktion des Wissens auf einem äußerst subjektiven und kreativen Bereich - der Modellbildung - beruhen. Der Übergang vom menschlichen „know-how“ zum „say-how“ bereitet dem Knowledge Engineer erhebliche Probleme (dieser Engpaß wird auch das „Feigenbaumsche Bottleneck“ genannt).

Andererseits lassen sich auf diese Weise wesentliche Probleme der Modellierung komplizierter Systeme, wie

- ungenügende A-priori-Information über das zu modellierende System, d.h. über die wichtigsten Einflußgrößen, funktionelle Abhängigkeit der Ausgangsgröße von den Einflußgrößen, Störgrößen;
- große Anzahl von Einflußgrößen, die nicht erfaßt werden können;
- inadäquate Beschreibungsmethoden für vorhandene Unbestimmtheit;
- kleine Stichproben mit erheblich verrauschten Daten nicht lösen. Ein möglicher Ausweg ist deshalb eine Wissensextraktion aus den Daten.

2. Wissensextraktion aus den Daten

Insbesondere bei ungenügender A-priori-Information, aber auch in den anderen weiter oben angegebenen Fällen ist es möglich, aus den Daten mathematische Modelle zu generieren, wobei sich die Einbeziehung der Experten auf die Bereitstellung geeigneter Algorithmen sowie auf die Vorgabe der Kriterien zur Auswahl der Modelle optimaler Kompliziertheit beschränkt. Wissenschaftliche Grundlagen einer derartigen Selbstorganisation mathematischer Modelle sind u.a.:

- Black-box- Methode (experimentelle Systemanalyse) als prinzipielle Herangehensweise an die Analyse von Systemen auf der Grundlage der Input-Output-Realisierungen;
- induktive Herangehensweise als Konzept der Herleitung allgemeingültiger Modelle aus den Realisierungen;
- Konnektionismus, d.h. die Darstellung komplizierter Funktionen mit Hilfe von Netzwerken elementarer Funktionen;
- Nutzung der kybernetischen Prinzipien der Selbstorganisation;
- Gödels Unvollständigkeitstheorem bzw. das von Tichonov entwickelte Prinzip der Regularisierung von ill-defined tasks.

Historisch haben sich auf dieser Grundlage die Theorie neuronaler Netze sowie die Theorie statistischer lernender Netze herausgebildet.

Während erstere zur Zeit in der Literatur eine große Beachtung findet und deshalb auch zunehmend bei der Modellierung eingesetzt wird, beschränken sich die Arbeiten zu letzterer auf einige wenige Arbeiten im Zusammenhang mit den GMDH algorithm. Das ist umso erstaunlicher, als die Nachteile der Theorie neuronaler Netze im Zusammenhang mit der Modellbildung auf der Hand liegen.

Ergebnis der Lernphase neuronaler Netze sind implizite Modelle. Sie ergeben sich aus der Architektur des erhaltenen neuronalen Netzes, der erlernten Parameter sowie der verwendeten Aktivierungsfunktionen (z.B. Sigmoidale). Nur mit großem Aufwand und näherungsweise (z.B. durch Reihendarstellung der sigmoidalen Aktivierungsfunktion) ist es möglich, die sich ergebende Modellstruktur explizit anzugeben (siehe z.B. [Otto,94]). Offensichtlich existiert für jede Aufgabenstellung eine effektive bzw. optimale Netzwerkarchitektur, die durch Anzahl der Schichten und Anzahl der Neuronen in jeder Schicht sowie die zulässigen Beziehungen zwischen den Neuronen bestimmt wird. Ihre effektive Wahl ist jedoch bislang weitgehend ungelöst, dementsprechend dominieren Heuristik und langwieriges Experimentieren, die in jedem Fall Erfahrungen im Umgang mit neuronalen Netzen voraussetzen. Zur Bestimmung der unbekannt Parameter der Neuronen finden rekursive Verfahren unter Verwendung iterativer, oftmals auch heuristischer Lernregeln Anwendung. Für diese ist in der Regel eine langsame Lerngeschwindigkeit kennzeichnend (bei Prognosemodellen nach [Rehkugler,91] mehrere Tage bis zu einer Woche bzw. eine sehr aufwendige Spezialhardware). Die auf diese Weise erhaltenen Ergebnisse hängen darüberhinaus von den gewählten Anfangswerten ab.

So ist die Entwicklung eines Prognosemodells auf der Grundlage neuronaler Netze ein zeitaufwendiger, experimenteller Prozeß, der erhebliche Kosten verursachen kann. Wie im Weiteren

zu zeigen ist, hat der zweite Weg - statistische lernende Netze - einige der genannten Probleme nicht bzw. bietet Lösungsansätze für deren Bewältigung.

3. Statistische lernende Netze und Besonderheiten der GMDH algorithm

3.1 Statistische lernende Netze

Die im Weiteren behandelten Algorithmen der induktiven Modellbildung gehen zurück auf die von A.G.Ivachnenko begründete Methode der gruppenweisen Berücksichtigung der Argumente (GMDH: group method of data handling). Sie stellt gegenwärtig eine der wesentlichen Säulen der Theorie statistischer lernender Netze dar.

Im Zusammenhang mit der empirischen oder induktiven Erzeugung von Funktionen vieler Variablen stellt ein Netzwerk eine Funktion dar, die sich durch Komposition aus vielen elementaren Funktionen (Basisfunktionen) ergibt. Diese Basisfunktionen stellen Elemente, Blöcke, Einheiten oder Knoten eines komplizierten Netzwerkes dar und entsprechen den Neuronen in neuronalen Netzen. In der Regel beschränken sie sich entweder auf nichtlineare Funktionen einiger weniger Variablen oder lineare Funktionen vieler Variablen. Die Bezeichnung lernende Netzwerke bedeutet, daß sich das Netzwerk die Funktion auf der Grundlage repräsentativer Beobachtungen der relevanten Variablen erlernt [Barron,88].

Kolmogorov und Lorentz haben mathematisch bewiesen, daß jede kontinuierliche Funktion $F(x_1, \dots, x_d)$ im d -dimensionalen Einheitsraum $[0,1]^d$ exakt als eine Komposition von Summen und kontinuierlichen eindimensionalen Funktionen dargestellt werden kann.

3.2 Besonderheiten der GMDH algorithm

1. Elementare Prozessoren

Offensichtlich realisieren die elementaren Prozessoren in einer gegebenen Schicht eine Transformation der zulässigen Inputs in einen Output. In Abhängigkeit von der A-priori-Information über die darzustellende Netzwerkfunktion $F(x_1, \dots, x_d)$ werden einfache Funktionen oder Operatoren gewählt, deren Kombination im Netzwerk die Darstellung der gewünschten Netzwerkfunktion garantiert. Diese durch die A-priori-Information bewußt beabsichtigte Einschränkung der Vielfalt möglicher Funktionen ermöglicht in endlicher Zeit die Netzwerksynthese abzuschließen.

Parametrische GMDH - Algorithmen enthalten Prozessoren mit 2 (manchmal auch 3) Inputs x_i , x_j , wobei in einer Schicht prinzipiell alle paarweisen Kombinationen zulässig sind. Es gilt $y = f(x_i, x_j)$, wobei f eine beliebige lineare oder nichtlineare Basisfunktion darstellt, die so gewählt wird, daß deren Komposition zur gewünschten Netzwerkfunktion führt, wie z.B.

$$f_1(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j$$

$$f_2(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2$$

$$f_3(x_i, x_j) = a_1 x_i + a_2 x_j x_k .$$

Komponenten des Input -Vektors x_i , $i=1,2,\dots,n$, können unabhängige Variable, Funktionen aber auch zeitverzögerte Realisierungen der Systemgrößen sein. Entsprechend der Wahl der Basisfunktionen f ergeben sich parametrische Modelle in Form von Differenzgleichungen, Polynomen, Trendfunktionen u.a.. Als Transferfunktion dient eine Signumfunktion; gefeuert wird, wenn das Selektionskriterium einen Schwellwert übertrifft.

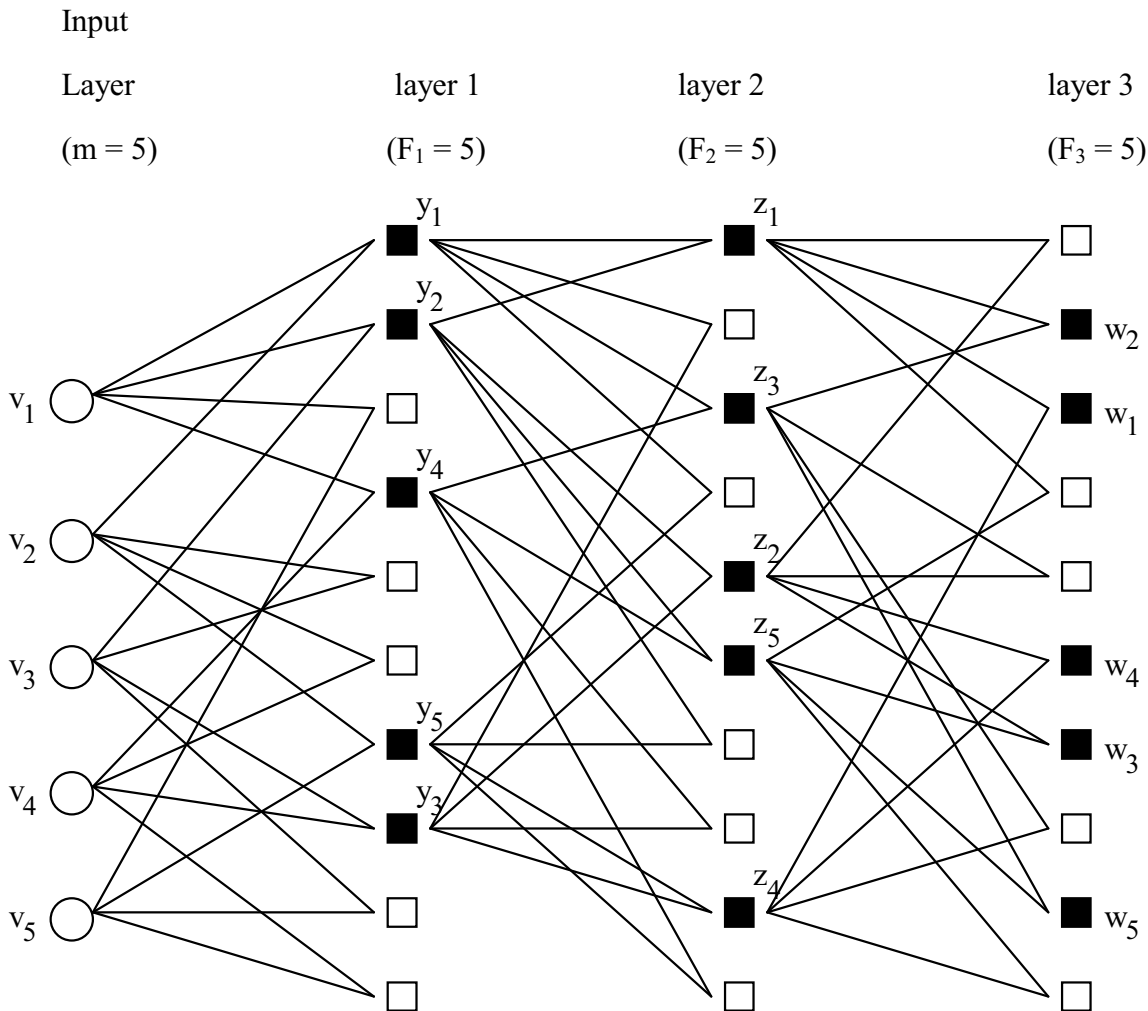


Bild 1: Netzwerk (3 Schichten) mit unvollständiger Induktion

Neben den sich auf diese Weise ergebenden parametrischen Algorithmen wurden auch nichtparametrische Algorithmen entwickelt, die eine Parameterschätzung nicht erfordern. Eine Begründung für eine solche abgeschwächte mathematische Beschreibung, d.h. für die Anwendung von Methoden, die das Untersuchungsobjekt auf einer weniger detaillierten Sprache als der Sprache der Differenzgleichungen (bzw. deren Lösung z.B. in Form von Polynomen) beschreiben, liefert das Prinzip der Adäquatheit des zu beschreibenden Systems mit seiner mathematischen Beschreibung nach Ashby und Beer [Ivachnenko/Müller, 93].

2. Netzwerkkonstruktion

Die Algorithmen der Selbstorganisation realisieren im Rahmen der Vorgabe der Schwellwerte bzw. der Anzahl der in jeder Schicht zu selektierenden Elemente und der Wahl der Selektionskriterien (möglichst aus inhaltlicher Sicht) eine adaptive Netzwerkkonstruktion: die Neuronen in jeder Schicht und

ihre Verbindungen zu den vorhergehenden Schichten ergeben sich während der Synthese, die Anzahl der Schichten ist im Vorhinein unbekannt. Offensichtlich können unter verschiedene Prinzipien zur Generierung alternativer Modellvarianten mit wachsender Kompliziertheit Anwendung fin-

den, unterschieden werden Algorithmen mit vollständiger bzw. unvollständiger Induktion. Für letztere gilt im einfachsten Fall (Verbindung lediglich zur benachbarten Schicht) folgendes: Die Anzahl der Inputs der 1. Schicht sei n . Von den $\binom{n}{2}$ möglichen Neuronen der 1. Schicht werden F_1 selektiert, ihre Outputs sind Inputs der sich damit ergebenden $\binom{F_1}{2}$ Neuronen der 2. Schicht usw. (Bild 1). Die F_i werden entweder vorgegeben oder ergeben sich aus gegebenen Schwellwerten.

3. Lernregeln

Anstelle rekursiver Verfahren zur Bestimmung der unbekannt Parameter der Neuronen unter Verwendung iterativer, oftmals auch heuristischer Lernregeln finden bei den GMDH algorithm direkte Schätzverfahren für jedes Neuron in jeder Generation Anwendung. Für ausgewählte Klassen von Basisfunktionen (z.B. in den unbekannt Parametern linear) können direkte Methoden zur Identifikation der unbekannt Parameter angewendet werden, wie z.B. die Maximum-Likelihood-Schätzung.

4. Äußere Ergänzung

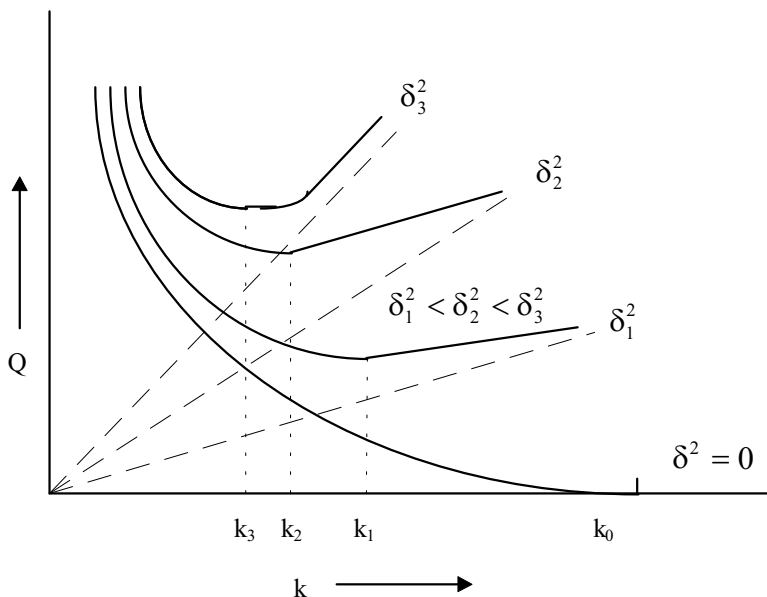
Die Aufgabe, lediglich auf der Lernfolge ein Modell optimaler Kompliziertheit (nonphysical model) auszuwählen, ist nicht korrekt gestellt. Zur eindeutigen Auswahl wird ein äußeres Kriterium verwendet, d.h. die Selektion erfolgt auf einer Prüffolge.

5. Einbeziehung der A-priori-Information

Das sicher vorhandene A-priori-Wissen der Experten findet bei der Auswahl der universellen Modellklassen und der alternativen, teilweise heuristischen Modellvarianten verschiedener Detailliertheits- und Beschreibungsebenen Anwendung. Der Anwender wählt entsprechend seinem inhaltlichen Anliegen und den von ihm verbal formulierten Anforderungen an ein Modell optimaler Kompliziertheit sein Selektionskriterium (Hierarchie von Selektionskriterien) aus. Es ist zu beachten, daß für jede Aufgabenstellung (empirischer Datensatz) und jede Menge von Basisfunktionen ein "bestes" Auswahlkriterium und eine "beste" Aufteilung der Beobachtungen existieren.

6. Nachweis der optimalen Kompliziertheit

Eine wichtige Eigenschaft aller Auswahlalgorithmen mit quadratischen Auswahlkriterien besteht darin, daß bei vorhandenem Rauschniveau (Einfluß stochastischer Störgrößen, fehlender aber auch redundanter Einflußgrößen) ein einziges Modell optimaler Kompliziertheit existiert. Mit wachsendem Rauscheinfluß verringert sich die optimale Kompliziertheit dieses Modells, d.h. die Modelle werden einfacher und damit gegenüber Störungen robuster (Bild 2).



4. SelfOrganize!

Anwendung fand das von DeltaDesign Software Berlin entwickelte Paket "SelfOrganize!" [Lemke, 94]. Hier wurde ein parametrischer GMDH-Algorithmus umgesetzt, der im Verlauf des Modellierungsprozesses zusätzlich zur adaptiven Netzwerksynthese eine Optimierung der Struktur der Übertragungsfunktion jedes einzelnen Neurons vornimmt, so daß die erhaltene Netzwerkfunktion eine Komposition unterschiedlicher, a priori unbestimmter Neuronen darstellt.

Bild 2: Einfluß des Rauschanteiles auf die optimale Kompliziertheit des Modells

Optimiert wird ein Polynom 2. Grades

$$f_2(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2$$

mit dem Ergebnis, daß die in Abhängigkeit vom gewählten Schwellwert generierten Neuronen im einfachsten Fall mit dem Absolutglied und im kompliziertesten Fall mit dem vollständigen Polynom beschrieben sein können. Das durch die erhaltene Netzwerkfunktion repräsentierte Zeitreihen- oder Input-Output-Modell steht unmittelbar nach der Modellierung explizit zur Verfügung und kann somit außer zur Vorhersage von Kennzahlen oder Kennzahlensystemen auch zur Interpretation und Analysetätigkeit herangezogen werden.

Die in den folgenden Beispielen erhaltenen Modelle wurden auf einem PC mittlerer Leistung innerhalb weniger Stunden erstellt.

4. Analyse von Kennzahlensystemen

Erfolgreiche Anwendungen der GMDH algorithm 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 Anwendungen zur Modellierung volkswirtschaftlicher, betriebswirtschaftlicher, ökologischer, meteorologischer u.a. Systeme. Ein wichtiger Anwendungsbereich insbesondere im Rahmen von entscheidungsunterstützenden Systemen, d.h. bei der kurzfristigen Bereitstellung zukunftsorientierter entscheidungs- und strategierelevanter Informationen, ist die Analyse und Vorhersage von Kennzahlensystemen. Kennzahlensysteme, d.h. die Zusammenstellung (und Verdichtung) von Kennzahlen, die in sachlogischer Beziehung stehen, dienen sowohl der externen Analyse (z.B. zur finanzwirtschaftlichen oder erfolgswirtschaftlichen

Bilanzanalyse), als auch der internen Analyse (z.B. zur Frühwarnung und Unterstützung von Planungs- und Kontrollprozessen mittels Kreditwürdigkeitsprüfung, Unternehmensbewertung u.a.) .

4.1 Vorhersage der Aktivitätsindizes der New Yorker Wertpapierbörse

t	x ₁	x ₂	x ₃	x ₄	t	x ₁	x ₂	x ₃	x ₄
65	1.09	0.17	0.76	11.80	68	0.65	0.52	0.29	4.87
66	0.99	0.56	0.16	12.00	69	0.87	0.83	0.90	0.73
67	0.58	0.55	0.16	8.04	70	2.12	0.65	0.90	0.98
					MA	1.05	0.55	0.53	6.40
					D				

Tafel 1 :Vorhersagefehler $\delta = \left| \frac{x_{it} - x_{ik}^M}{x_{it}} \right| 100\%$ der Aktivitätsindizes (New Yorker Wertpapierbörse)

Gegeben sind 4 Variable, die die Aktivität der New Yorker Wertpapierbörse vom 11.5.-17.8.1991 (70 Tage) kennzeichnen: x₁ - minimaler Wert des Aktivitätsindex, x₂ - maximaler Wert des Aktivitätsindex, x₃ - Wert des Aktivitätsindex zum Börsenschluß und x₄ - Summe der täglichen Operationen. Im Ergebnis der Modellierung mit 65 Realisierungen wurden die in der Tafel 1 dargestellten Vorhersagefehler erhalten.

4.2. Vorhersage betriebswirtschaftlicher Kenngrößen in einem Unternehmen

Gegeben sind aus einem Unternehmen der Elektrobranche 17 Kenngrößen x_i (i= 1, 2, ..., 17) mit jeweils 24 Realisierungen. Auf der Grundlage von 22 Realisierungen wurde vom Computer ein lineares, in sich widerspruchsfreies Differenzgleichungssystem der Art

$$\underline{x}_t = A \underline{x}_t + B_k \underline{x}_{t-k}$$

erstellt. Das Modell erlaubt analytische Aussagen, z.B. zum Einfluß der einzelnen Einflußgrößen und ihrer Verzögerungen. Für das hier betrachtete Unternehmen sind die auf der Basis des generierten Gleichungssystems erhaltenen Vorhersagefehler für einen Vorhersagezeitraum von 2 Jahren in Tafel 2 enthalten.

	x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈	x ₉
1	3,09	4,70	0,68	0,11	6,07	0,60	2,26	3,04	11,02
2	14,28	8,26	7,05	6,59	12,62	0,96	1,12	0,68	18,41
MAD	8,68	6,48	3,87	3,35	9,35	0,78	1,69	1,86	14,72

	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}
1	15,16	4,20	7,09	3,28	5,83	8,62	0,88	7,34
2	27,48	14,51	3,82	1,46	4,41	20,12	0,76	19,08
MAD	21,32	9,35	5,45	2,37	5,12	14,37	0,82	13,21

Tafel 2: Vorhersagefehler (MAD [%]) betriebswirtschaftlicher Kenngrößen

4.3 Ermittlung der Kreditwürdigkeit

Gegeben sind 19 Kenngrößen von 81 Unternehmen, die von einer Großbank zur Kreditentscheidung zu Grunde gelegt wurden. Die Prüfung bezog sich dabei auf die Kriterien Vermögenslage (Vermögensstruktur, Kapitalstruktur), Liquidität (aktuelle Liquidität, Finanzplananalyse) und Rentabilität (siehe hierzu [Bischoff, 91]). Von den 81 Fällen wurden 10 Entscheidungen als Testfälle (5 positive, 3 negative, 1 unentschiedene Entscheidung sowie 1 Grenzfall) vorgegeben. Die übrigen 71 Entscheidungen (36 negativ, 35 positiv) dienen als Lernfälle.

Erstellt wurden lineare und nichtlineare Modelle der Entscheidungsvariable von den 19 Kennzahlen aber auch zwei separate Gleichungssysteme zur Beschreibung der negativen bzw. positiven Fälle. Beide Herangehensweisen liefern Modelle, die Aussagen über den Einfluß der einzelnen Kennzahlen auf die jeweilige Entscheidung liefern.

Übereinstimmend mit den in [Bischoff, 91] erhaltenen guten Ergebnisse für die verschiedenen neuronalen Netze wurden in der Regel die 5 positiven und 3 negativen Entscheidungen bestätigt und der Grenzfall ebenfalls als positiv angesehen. Der unentschiedene Fall wurde überwiegend positiv entschieden. Im Unterschied zu neuronalen Netzen liefert diese Herangehensweise unmittelbar die Erklärungskomponente, die in [Bischoff, 91] als entscheidendes Akzeptanzproblem der neuronalen Netze für den Kreditsachbearbeiter angesehen wird.

Literatur

Barron, A.R., Barron,R.L.:Statistical learning networks: a unifying view. Proceedings of the 20th Symposium Computer Science and Statistics, 1988.

Bischoff,R., Bleile, C., Graalfs, J.: Der Einsatz Neuronaler Netze zur betriebswirtschaftlichen Kennzahlenanalyse. Wirtschaftsinformatik 33 (1991) H.5

Lemke, F.:SelfOrganize! - software tool for modelling and prediction of complex systems. Systems Analysis Modelling Simulation 11 (1994) H.6

Ivachnenko, A.G., Müller, J.-A.: Self-Organization of optimum physical clustering of a data sample for a weakened description and forecasting of fuzzy objects. Pattern Recognition and Image Analysis 3 (1993) H.4

Otto, P.: Identifikation nichtlinearer statischer und dynamischer Systeme mit Künstlichen neuronalen Netzen. 39.IWK Band 3. Ilmenau 1994

Rehkugler, H.: Künstliche Neuronale Netze in der Finanzanalyse: Eine neue Ära der Kursprognose? Wirtschaftsinformatik 33 (1991) H.5