FINANCE & TECHNOLOGY

Journal of Computational Intelligence in Finance

Self-Organizing Data Mining For A Portfolio Trading System

Frank Lemke and Johann-Adolf Mueller

This paper describes the application of data mining algorithms for a portfolio trading system. The goal of data mining in this case is the prediction of assets of a portfolio by means of parametric or non-parametric models. Parametric models are adaptively created from data by the Group Method of Data Handling (GMDH) in the form of networks of optimized transfer functions. Non-parametric models are selected from a given variable set by analogues complexing, representing one or more patterns of a trajectory of past behaviour "which are analogous to a chosen reference pattern. Approaches to self-organizing modelling include not only core data mining algorithms but also an iterative process of generating alternative models with growing complexity, along with their evaluation and validation, and the selection of a model with optimal complexity. In this paper, these approaches are denoted as self-organizing data mining.

In a modelling/prediction module, self-organizing data mining is performed for the purpose of extracting and synthesising hidden knowledge from data systematically and quickly. The control module of the trading system is responsible for signal generation based on predictions provided by the modelling module. Initial performance results of a trading system are presented. The trading system simulates trading a portfolio of diverse stocks using daily out-of-sample price data.

1. INTRODUCTION: DATA MINING AND SELF-ORGANIZING MODELLING

Decision making in finance, such as trading a portfolio of diverse assets, requires sufficient problem definition and motivation. Information technology delivers a flood of data to decision makers, resulting in questions on how to leverage them. Data mining techniques and tools can assist humans in analysing the mountains of data and to turn information located in the data into successful decision making. Data mining is an interactive and iterative process of numerous subtasks and decisions such as data selection and pre-processing, the application of data mining algorithms and analysis of the extracted knowledge.

For sophisticated data mining applications, it is important to try to limit the involvement of users in the overall data mining process to the inclusion of existing *a priori* knowledge while making this process more automated and objective. The primary interest of most users of data mining applications (that is, those who may lack sufficient knowledge of mathematical, cybernetic and statistical techniques or sufficient time for dialog driven modelling tools) are results. Self-organizing modelling is based on these demands and is a powerful way to generate models of ill-defined problems. Figure 1 illustrates a spectrum of self-organizing modelling. The goal of self-organizing modelling is to perform data mining by generating mathematical models from empirical data more or less automatically. Such processing is denoted in this paper as self-organizing data mining. Associated mathematical models include the following:

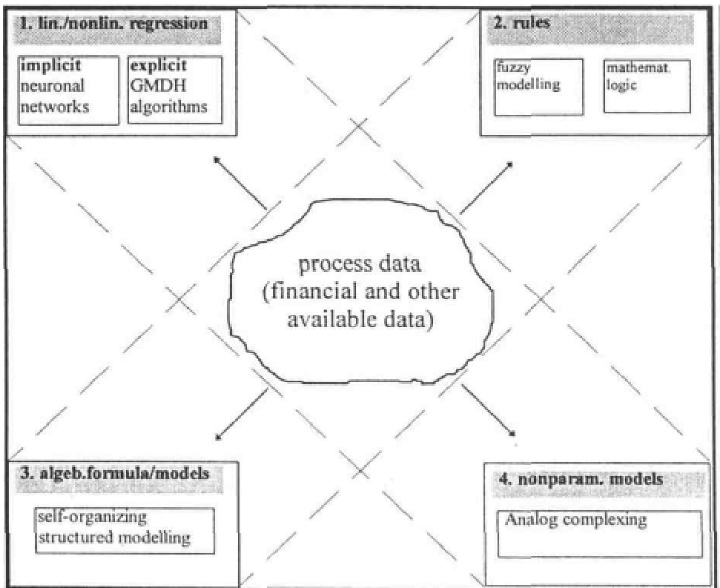


FIGURE 1. Spectrum of self-organizing modeling.

1. regression functions
Commonly, statistically-based principles are used to select parametric models. In addition to sophisticated methods of mathematical statistics, there has been much publicity regarding the ability of artificial neural networks to learn and to generalize.

A second regression-based method for model self-organization is the Group Method of Data Handling (GMDH). Using the GMDH algorithm, parametric models, such as time-series models, in the form of multi-input/single-output and multi-input/multi-output systems (that is, systems of equations) are adaptively created from data in the form of networks of optimized transfer functions (Active Neurons). In contrast to neural networks, the GMDH algorithm works on an important additional principle: that of *induction.* In this case, tree-like network is grown from seed information (input and output variables) from a simple single individual (neuron) in an evolutionary fashion using pairwise combination and survival-of-the-fittest selection to a desired final, not over-specialised behaviour (model). Neither the number of neurons and layers in the network, nor the actual behaviour of each created neuron (transfer function), are predefined. Instead, these are adjusted during the process of self-organization.

2. *rule-based models in the form of binary or fuzzy logic*
Rule induction from data uses genetic algorithms where the representation of models is in the familiar disjunctive normal form as discussed in Goonatilake [1994]. A self-organizing fuzzy model using the GMDH algorithm may be more useful for ill-defined financial applications, e.g., the generation of fuzzy trading signals from a given portfolio of data.

3. complex structured process models
Self-organizing structured modelling uses a symbolic generation of appropriate model structure (algebraic formula or complex process model) and the optimization or identification of a related set of parameters by means of genetic algorithms. This approach assumes that the elementary components are predefined and suitable for genetic coding.

4. non-parametric models (patterns) Non-parametric models are selected from a given variable set by *analogues complexing* representing one or more patterns of a trajectory of past behaviour which are analogous to a chosen reference pattern. Analogue complexing is based on the assumption that typical situations (behaviours) exist, i.e., each period of state evolution for a given multidimensional time process may have one or more analogues in history. If this assumption is correct, it is likely that a prediction could be obtained by transforming the known continuations of the historical analogues. It is essential that the search for analogous patterns is not restricted to a single state variable (time series) but includes a set of representative variables simultaneously and objectively. In fact, analogue complexing applied to financial applications could be considered as a kind of multidimensional, automated and objective chart analysis.

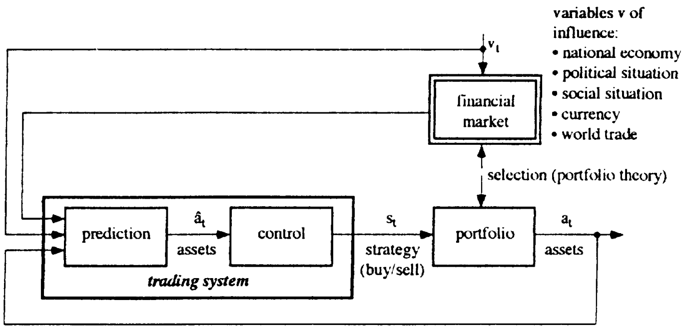


FIGURE 2. Predictive control of a portfolio.

This paper describes the synthesis of parametric models adaptively created using GMDH (3.2.1) and non-parametric models selected from a given variable set using analogue complexing (3.2.2) as part of a data mining approach for financial applications.

2. SELF-ORGANIZING DATA MINING FOR A PORTFOLIO TRADING SYSTEM

Trading systems may be used to generate trading signals on the basis of predictions of assets of a given portfolio. Such a predictive system is shown in Figure 2. Self-organizing data mining with respect to a portfolio trading system involves the automatic selection of useful knowledge for the purpose of generating trading signals for each portfolio contract based on available information. This information might include time series of variables of the national economy, political and social situations, currency, world trade, etc.

Generally, our trading system is split into 2 parts: (1) the prediction process and (2) the control process. The first is responsible for modelling the whole portfolio as part of a very noisy dynamic process and for predicting its time-dependent evolution. The control process involves transforming the predictions obtained by the modelling/prediction component into trading signals. Figure 3 outlines the basic structure of a trading system utilising a self-organizing data mining approach.

3. SELF-ORGANIZING DATA MINING

3.1 DATA SELECTION

3.1.1 INCLUSION OF *A PRIORI* INFORMATION

Applications of self-organizing modelling for describing complex systems have shown that a purely automatic approach to self-organizing models is not necessarily the most promising approach. The accumulation of information important to decision making — in our case, informa-

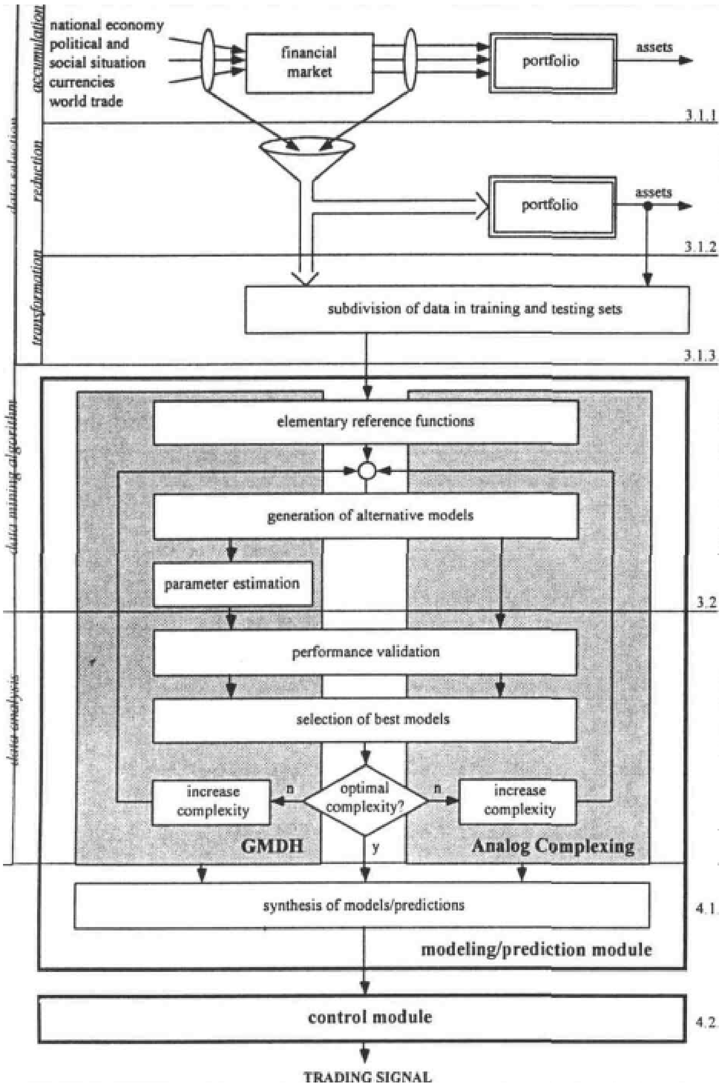


FIGURE 3. Self-organizing data mining for a trading system.

j tion important to the prediction of portfolio assets and the generation of trading signals — is not a formal task but, rather, a specialized financial task. Therefore, it is advisable to use *a priori* information about the system to be modelled. This includes any knowledge about input-output

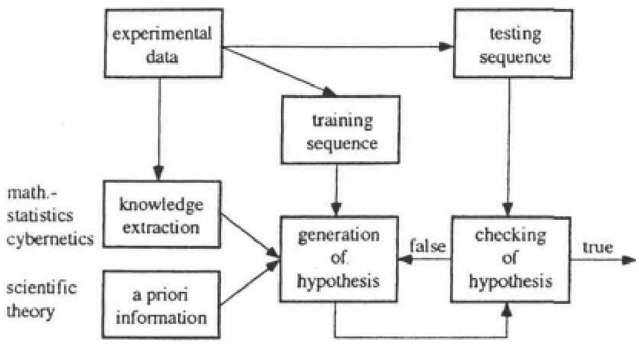


FIGURE 4. Basic scheme of self-organizing modelling for the case which considers *a priori* information.

relationships, structural information and causal relations well-known in economics, including knowledge accumulated from systems research by modelling large-scale systems (e.g., analysis of structure, stability, sensitivity and diagnosis of multicollinearity). This approach, illustrated in Figure 4, expands upon the basic scheme of self-organizing modelling.

3.1.2 REDUCING THE VARIABLE SET

As a first task, it is necessary to select a set of variables (available information) which may have influence on the evolution of the portfolio assets and which can be observed, measured or transformed using numerical values. A next, very important and high-priority step is the objective selection of essential variables. This is equivalent to analyzing the observability and controllability of a given dynamic system, i.e., to determining the necessary dimension of the state space required to describe the system.

In the theory of self-organizing modelling, the so-called algorithm of Objective Systems Analysis (OSA) has been developed and implemented, and is described in more detail in Madala and Ivakhnenko [1994]. The purpose of OSA is to assist in the selection of not only the model of optimal complexity (in the form of a single equation) but also the optimal number of equations necessary for describing the given dynamic system as a whole.

Another approach to reducing the variable set is the creation of a model nucleus according to methods presented in Ivakhnenko and Mueller [1992]. A model nucleus represents a subset of significant variables which specify the essential dynamic behaviour of the whole system. All remaining variables are only a static derivation of the nucleus. The nucleus is created automatically by applying the algorithm of OSA to samples of the data (see Madala and Ivakhnenko [1994]). Using selection-type GMDH sorting algorithms, the optimal number and width of clusters for partitioning the variables can be derived.

Finally, the GMDH algorithm itself provides an effective method for selecting essential variables by utilising linear model construction techniques. Here, variables selected for the last layer in the model indicate an ensemble of variables for analyzing consistent patterns in the data. All three approaches of data reduction can be realized automatically.

3.1.3 Data Transformation

One important aspect of self-organizing modelling involves the use of external information (i.e., information not used for creating a given model) for the purpose of objectively selecting a model of optimal complexity. The basic scheme of self-organizing modelling is shown in Figure 4.

The typical approach to generating external data is to subdivide the available dataset into separate sets, one of which is retained as an out-of-sample set. The data can be divided either *explicitly* by using one of various splitting rules (which reduces the length of the available training dataset) or *implicitly* by using cross-validation methods.

Another way to obtain external information is to generate a testing set by adding artificial noise (randomisation) to the training set or by discretizing the training set.

3.2. SELF-ORGANIZING DATA MINING ALGORITHMS

Self-organizing data mining algorithms perform automatically, in an objective way, using the following steps (see Figures 3 and 5):

A. the generation of alternative models with different variables, which results in growing the complexity in each layer,
B. for parametric models, the estimation of unknown parameters of a training set, and the validation of performance based on a testing set as measured by at least two different criteria,
C. the selection of best models in each processed layer based on external information, and
D. as long as performance is improving, the growing of complexity in each layer is repeated, otherwise a final model of optimal complexity is selected.

The above sequence of steps guarantees the objectivity of the selection process. Its advantages are primarily noticed in the modelling of large, complex systems with many variables (>50). In our trading system, we performed self-organizing modelling for data mining using two separate techniques:

1. GMDH-type neural networks and/or
2. analogue complexing.

3.2.1 GMDH-TYPE NEURAL NETWORKS

The traditional GMDH algorithm was developed by A.G. Ivakhnenko in 1967 and was described by Madala and Ivakhnenko [1994]. A tool which implements GMDH and analogue complexing methods presented in this paper is described in Lemke [1995]. The GMDH algorithm utilized here generates an optimized transfer function and structure for each neuron. This results in a synthesized network that is composed of different, non-pre-defined neurons and their corresponding transfer functions selected from all possible linear or non-linear polynomials described as:

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

The algorithm ensures that essential, independent variables will be selected and that descriptions of optimal transfer functions will be obtained. We note that, even though non-linear models are permissible, a linear model may be selected as optimal.

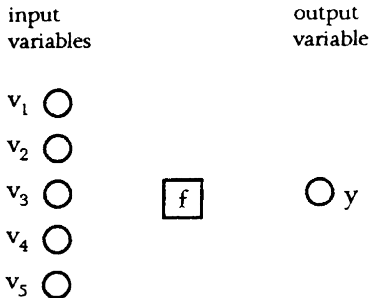
GMDH objectively selects the model of optimal complexity using an inductive approach. This includes the use of the cross-validation methods as an alternative to explicit data subdivision. Figure 5 illustrates the creation of a model of optimal complexity using a multilayered GMDH algorithm. Additionally, this inductive approach enables GMDH to overcome some of the most important problems commonly connected with the design, development and application of neural networks. For trading systems these are:

- Models obtained by most neural networks are models with information implicitly distributed over the network (e.g., the internally-connective representation of knowledge). This means that the knowledge which resides in such networks is inherently hidden. Thus, they cannot be used for data analysis, interpretation, validation or explanation tasks unless one uses dedicated knowledge extraction tools.
- With neural networks, developers estimate the structure of a network by choosing the number of layers in the network as well as the number of nodes and the types of transfer functions. This requires some knowledge of the theory of neural networks and experience in developing networks. The internal representation of knowledge does not support this trial-and-error process.
- Neural networks are a kind of statistical estimation, which are sometimes implemented using algorithms that are slower and less effective than algorithms provided with statistical software.
- If noise is considerable over a short data sample, these models may become overfitted multivariate regression functions. However, given commonly-employed strategies for avoiding overfitting (such as validation using out-of-sample datasets), this may not be a problem. Still, such strategies do not necessarily lead to an optimally complex model.

As an example, we will describe the process for predicting BMW stock prices over one cycle using GMDH modelling and analogue complexing. As a result of the data selection process, 100 daily closing prices for 10 variables from the German stock market were chosen. These include the dollar exchange rate, the stock prices for BMW, VW, AUDI, Ford, and Porsche, the following stock indexes: DAX, FAZ, and the following financial characteristics: the Discont and Lombard rates. Prices for all of these covered the period of August 5, 1995 through December 11, 1995.

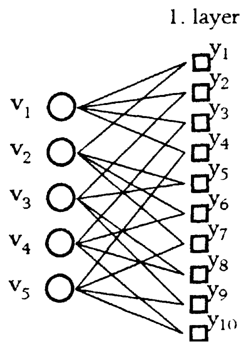
From the above data, 90 observations covering the period of August 5, 1995 through November 27, 1995 were used to generate models for predicting variables over the

I) before modelling



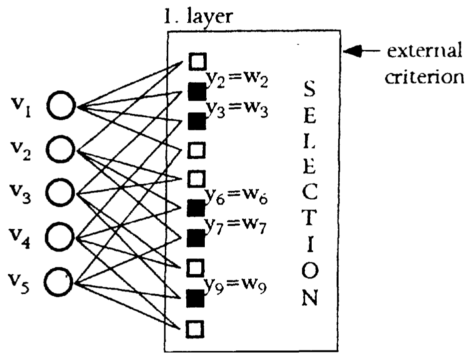
0 - vector of measured or synthesized data
f - H algorithm for creation of optimized transfer functions and of a networkfunction with increasing complicatedness

II) after creation of all models of the 1 st layer



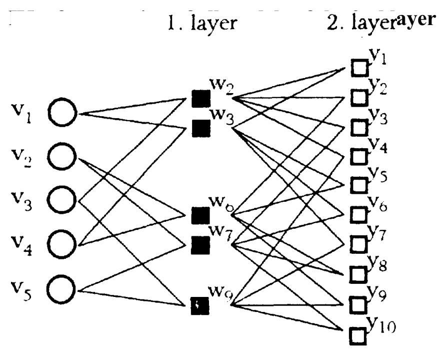
□ created neuron with an optimized transfer function
y_k=$f_k(v_i,v_j)$ (Active Neuron) order of regression model
y_k: ≤2 number of variables v in model y_k: ≤*2*

III) after selection of a number of best models



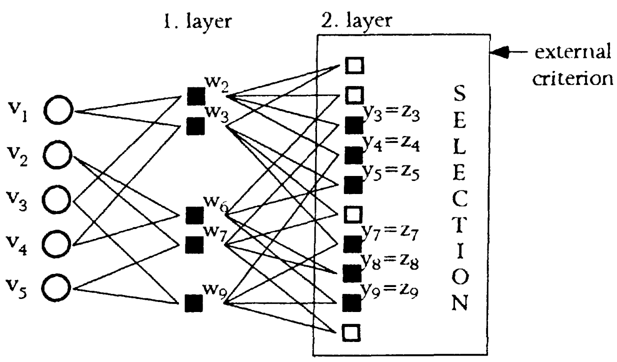
■ selected neuron

□ not selected neuron



• selected neuron of 1. layer with unchanged transfer function
□ created neuron with an optimized transfer function
y_k==$f_k(v_i,v_j)$ (Active Neuron) order of regression model
y_k: ≤4 number of variables v in model y_k: ≤4

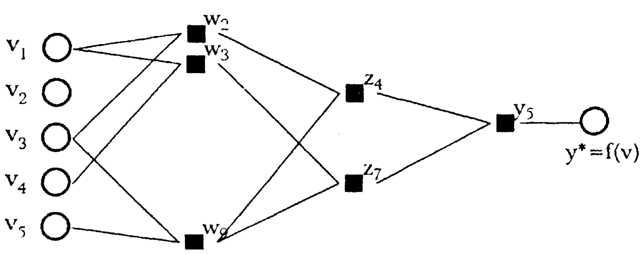
V) after selection of a number of best models



■ selected neuron

□ not selected neuron

VI) after self-inducted stop of modelling (here: after 3 layers) and selection of a best model y*



explicit analytical available optimal complex model. order of regression model: ≤8

FIGURE 5. Creation of an optimal complex model using a multilayered GMDH algorithm.

date	observation	prediction	error (MAD[%])
November 28, 1995	785	781.9	0.39
November 29, 1995	781.5	780.4	0.13
November 30, 1995	780	777.8	0.27
December 1, 1995	784	773.5	1.35
December 4, 1995	782	774.5	0.97
December 5, 1995	777	771.4	0.72
December 6, 1995	768	765.5	0.73
December 7, 1995	763	761.7	0.18
December 8, 1995	750	758.7	1.15
December 11, 1995	752	760.3	1.11

TABLE I. Prediction error for BMW.

period of November 28, 1995 through December 11, 1995. Normalization and denormalization of the data, as an integrated part of the overall modelling algorithm, were performed. The input variable set consisted of the above variables and their lagged samples (up to a lag of 15) for a total of 159 inputs. The output variables included each of the 10 original variables. The complete modelling process results in the creation and validation of thousands of different models of increasing complexity.

The following optimal model was generated for BMW stock prices:

$$\begin{aligned}&\text{BMW/} == 59.16\ 4\text{-}\ 0.776\text{BMW}_{t-1}\text{-}\ 73.632\text{Dollar}_{t-i}\\ &- 0.231\text{Ford}_{t-2} + 0.135\text{Ford}_{t-6} + 0.672\text{VW}_t\\ &- 0.472\text{VW}_{t-1}\end{aligned}$$

where the relation: BMW/ = f{BMW_{t-1}, Dollar_{t-1}, $\Delta_1\text{VW}_t$, $\Delta_4\text{Ford}_{t-2}$} was considered important for further analysis. The predictions for BMW prices are shown in Figure 6. Prediction errors based on performance measured using mean absolute deviation (MAD) are shown in Table 1. MAD is defined as:

$$\text{Dollar}_t = 0.364 + 0.906\text{Dollar}_{t-1} -0.0002\text{Ford}_{t-2} + 0.0006\text{FAZ}_t - 0.0005\text{FAZ}_{t-1} - 0.0003\text{FAZ}_{t-2} + 0.0001\text{FAZ}_{t-11}$$

$$\text{VW}_t = - 48.728 + 0.87\text{VW}_{t-1} - 0.091\text{VW}_{t-3} + 0.078\text{DAX}_t - 0.03\text{DAX}_{t-1} - 0.057\text{BMW}_{t-1} + 0.173\text{Audi}_{t-9}$$

$$\text{Audi}_t = 10.5 + 0.17\text{Audi}_{t-2} + 258.51\text{Dollar}_{t-1} + 109.7\text{Dollar}_{t-5} - 38.32\text{Dollar}_{t-8} - 0.03\text{DAX}_{t-2}$$

$$\text{FAZ}_t = 299.7 + 0.29\text{DAX}_t - 0.05\text{DAX}_{t-5} + 0.11\text{Porsche}_t + 0.06\text{BMW}_{t-13} - 0.178\text{Ford}_{t-8}$$

$$\text{DAX}_t = 1420.8 + 0.63\text{DAX}_{t-1} + 232.3\text{Dollar}_{t-1} - 0.98\text{FAZ}_{t-2} + 0.84\text{VW}_{t-1} - 0.23\text{Ford}_{t-2} - 0.405\text{Ford}_{t-7}$$

$$\text{Discont}_t = -2 + \text{Lombard}_t$$

$$\text{Lombard}_t = 0.29 + 0.947\text{Lombard}_{t-1}$$

$$\text{Ford}_t = 770.04 + 0.35\text{Ford}_{t-3} + 0.18\text{Ford}_{t-4} - 13.08\text{Discont}_{t-3} - 57.86\text{Lombard}_{t-15}$$

$$\text{Porsche}_t = - 122.66 + 0.586\text{Porsche}_{t-1} + 154.8\text{Dollar}_{t-1} + 0.077\text{DAX}_{t-15}$$

TABLE 2. System of equations obtained using techniques presented in the text.

$$\text{MAD} = \frac{1}{P}\sum_{i=1}^{P}\left|\frac{y_i - \hat{y}_i}{y_i}\right| * 100\%$$

where v, represents the actual prices, y, represents the predicted prices, and P represents the forecast horizon. Results for the other 9 output variables are presented in Table 2.

Table 3 shows the prediction errors for all 10 variables over 5 and 10 day periods. Similar results were obtained for other days using updated models. From an analysis of these models, we note that, as expected, their structure changes due to varying relationships between variables.

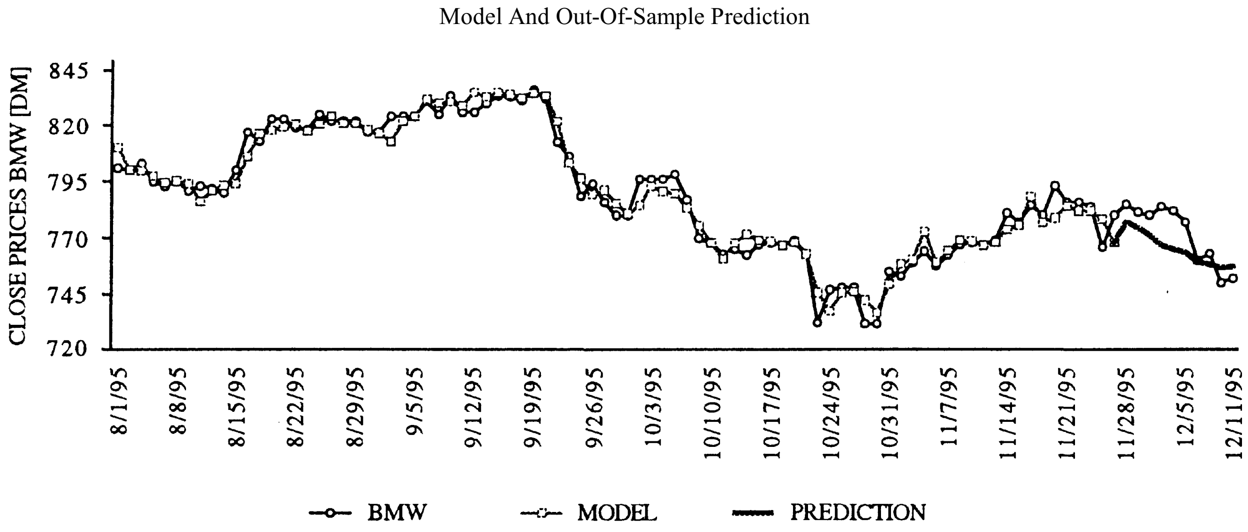
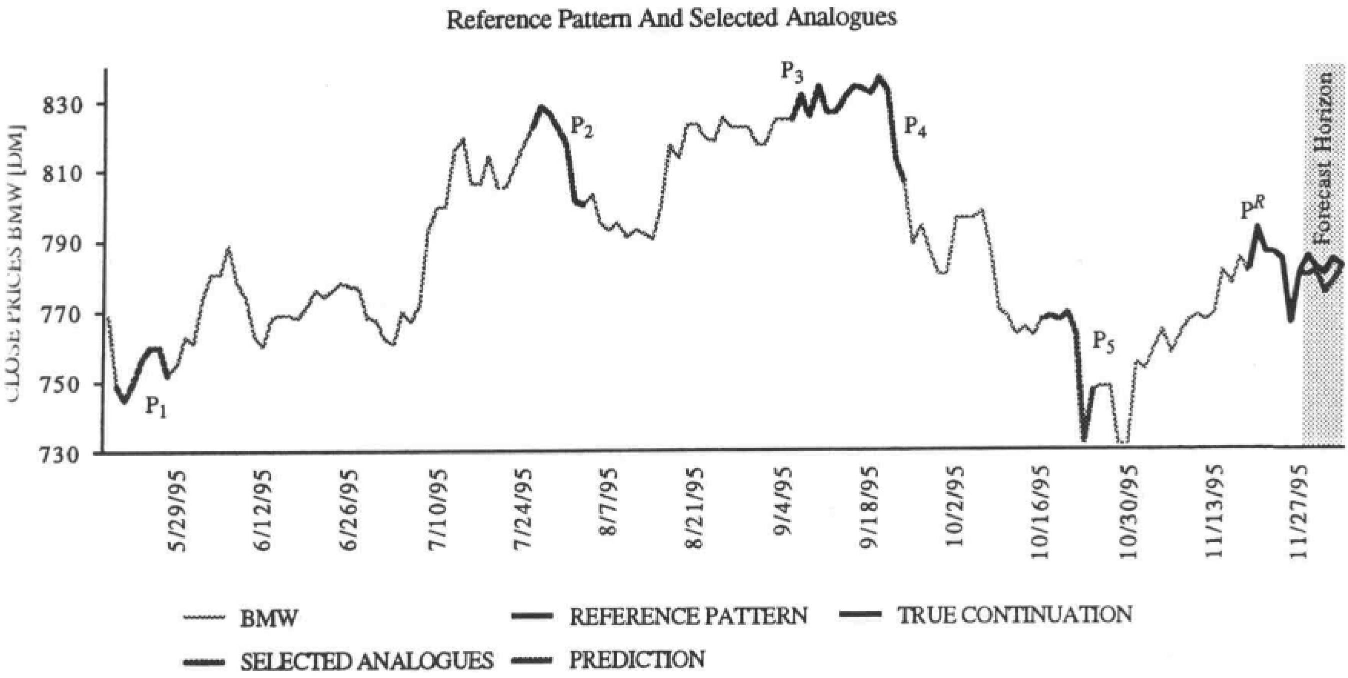


FIGURE 6. Graph of the model and 10-day, ex ante predictions for BMW,



FIGURE?. Reference pattern and selected analogous patterns for a one-dimensional process.

3.2.2 ANALOGUES COMPLEXING

Analogues complexing was developed by Lorence [1969] and was first successfully applied to meteorological forecasts. Within the past few years, it has been enhanced using inductive, self-organizing techniques and an advanced selection procedure to make it applicable to evolutionary processes. In analogue complexing, the observed process itself is used for forecasting. Forecasts are not calculated in the classical sense but selected from a table of observed data. This method is denoted as non-parametric because there is no need to estimate parameters. The main assumptions are:

- the system to be modelled is described by a multi-dimensional process,
- many observations represented by sampled data are available,
- the multi-dimensional process is sufficiently represented by the sample of observations, and
- it is possible that past behaviour will repeat in future.

If we succeed in finding past patterns which are analogous to the most recent (reference) pattern, predictions can be achieved by applying the known continuation of the analogous patterns to the reference pattern. However, this relationship, by itself, is only useful for non-evolutionary processes.

We will now describe an inductive search method that is suitable for forecasting multi-dimensional evolutionary processes, such as financial processes. In conjunction with this goal, a question arises regarding whether or not it is even possible to successfully apply such a method to forecasting evolutionary processes. Other than the philosophical side of the question, there is the need for formalizing the problem. For instance, for evolutionary processes,

$$P_k(i) = \begin{bmatrix} x_{1i} & . & . & x_{li} & . & . & x_{mi} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ x_{1i+j} & . & . & x_{li+j} & . & . & x_{mi+j} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ x_{1i+k-1} & . & . & x_{li+k-1} & . & . & x_{mi+k-1} \end{bmatrix}$$

stationarity, as one important condition of this methodology, is not fulfilled.

If it is possible to estimate the unknown trend (and perhaps the seasonal effects), the difference between the process and its trend can be used for analogue complexing.

	DOLLAR	BMW	vw	AUDI	FORD	PORSCHE	FAZ	DAX	DISCONT	LOMBARD
5 DAYS	1.06	0.62	3.47	1.47	0.54	2.53	1.14	2.00	0	0
10 DAYS	1.32	0.70	5.24	3.04	0.58	2.45	1.75	2.91	0	0.5

TABLE 3. Mean long-term prediction errors, MAD[%].

However, since the trend is an unknown function of time, the subjective selection of an appropriate function is a difficult problem. One objective solution is made available by application of the GMDH algorithm through its extraction and transformation capabilities. Because of the dependency of analogue complexing on the selected trend function, it was decided to consider an alternative method, which is described in the following 4-step procedure:

7. *Generation of alternative patterns* For a given real-valued m-dimensional time series $x_t = \{x_{1t}, ..., x_{mt}\}$, $t = 1,2, ..., N$, with N number of observations, the pattern is defined as a table $P_k(i)$ of k rows (observations), starting at time i where k is the pattern length and / = 1, 2, ..., N-k+1 and m is the number of columns (variables):
Using a sliding window to generate the set of possible patterns $\{P_k(i), i=1(1)N-k+1\}$, we compared all possible patterns using a certain similarity measure (defined in step 3), which was chosen in advance with respect to the most recent reference pattern Pr = P_k=(N-k+1). Figure 7 shows the BMW time series (a one-dimensional process), the reference pattern, and five selected analogous patterns with a pattern length of 7.

2. Transformation of analogues
For the given reference pattern with k observations, one or more analogues patterns may exist in history: $Pk(i)|i \in J$, where J is the set of best analogues. Financial processes are evolutionary processes. Therefore, any

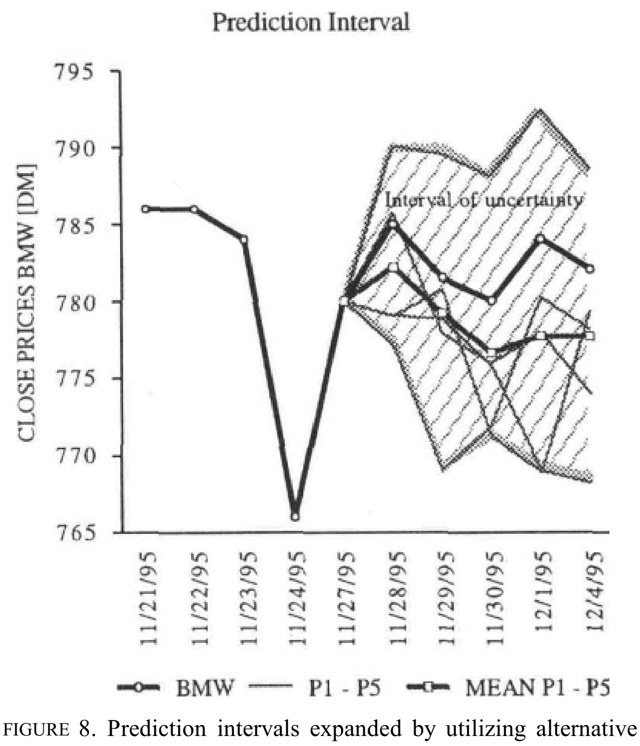


FIGURE 8. Prediction intervals expanded by utilizing alternative models.

similar patterns representing those processes may have different mean values, standard deviations and trends. This difference can be described by a transformation T,. Therefore, we have to select the most similar pattern in the past while similarity must be measured between the reference pattern and $T_i\,[P_k(i)|i \in J]$.

There exist several functions to describe the transformation I,. For a local approximation (with small k) of the unknown transformation T,, it is advisable to consider transformed patterns $T_i\,[P_k(i)]$ as a linear function of the pattern $P_k(i)$. This can be described as:

$$T_i[P_k(i)] = \begin{bmatrix} x^*_{1i} & . & . & x^*_{li} & . & . & x^*_{mi} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ x^*_{1i+j} & . & . & x^*_{li+j} & . & . & x^*_{mi+j} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ x^*_{1i+k-1} & . & . & x^*_{li+k-1} & . & . & x^*_{mi+k-1} \end{bmatrix}$$

with $x^*_{li+j} = a^i_{0l} + a^i_{1l} x_{li+j}$,
where $j = 0, 1, \ldots, k-1$; $i = 1, 2, \ldots, N-k+1$; and $l = 1, 2, \ldots, m$.

To explain such a transformation, let Xi be the Dollar exchange rate, let the reference pattern consist of 5 samples in December, and let an analogous pattern have 5 samples in September. In this case, the transformation has the following interpretation: the reference pattern of December are equal to a_0 plus a proportional part (a_1) that is equal to the analogous pattern of September.

Therefore, a'_{0l} can be interpreted as the difference between the states (here between the prices in December and September) and a'_{1l} can be interpreted as an extension factor. The unknown weights, a'_{0l} and a'_{1l}, for each pattern, can be estimated by means of the least squares method which provides not only the unknown weights but also the **total sum of squares.** This provides us with a similarity measure as described below.

3. Selection of the most similar analogues Distances can be measured by several similarity measures such as the Euclidean distance of points in the reference pattern and an analogue or by other measures of distance. In our case, it is possible to use as a similarity measure the **total sum of the squares** obtained by least squares estimation of unknown parameters, which gives us information regarding the similarity between the two patterns.

The algorithm required for selecting analogous patterns is now described. For the given reference pattern, it is necessary to select the most similar pattern $P_k(i)|i \in J$. The selection task is a four-dimensional problem with the following dimensions:

MACD Indicator - Principle

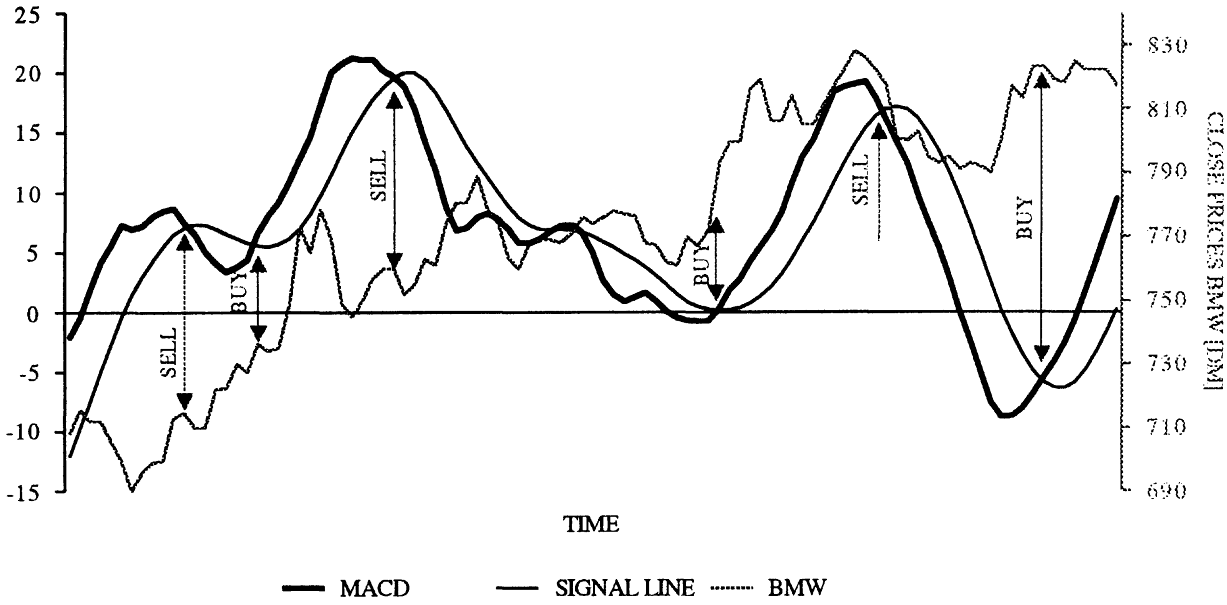


FIGURE 9. Principle of the MACD indicator.

- the set of variables used,
- the number of analogues selected,
- the length k of the pattern (number of lines used for each pattern), and
- the values of weight coefficients with which patterns are "complexed".

Cross-validation as well as different approaches to discretizing the samples can be used as necessary to generate external information.

4. Combining forecasts
Each selected analogue is used to provide a forecast by using continuation methods. These forecasts are then combined. Methods for combining forecasts have been described in Cheng et al. [1996] and Wu and Rehfuss [1997].
The unknown predictions $x_{N+i} = \{x_{1N+i}, x_{2N+i}, x_{mN+i}\}$, $i = 1,2,...,\tau$, where τ = forecast horizon of the m systems variables can be assumed to be a linear combination of the continuations of selected analogous patterns, i.e.:

$$x_{N+i} = g_0 + \sum_{j \in J} g_j x^T_{j+k+1} \text{ ,i = 1,2,...,}\tau$$

The unknown parameters go, gj, $j \in J$ can be estimated by means of parametric selection procedures such as the GMDH algorithm. The only problem is the small number of observations available for estimating the unknown parameters.
Therefore, for very small pattern lengths (k<< 10), it is advisable to use the weighted mean of the selected continuations

The weights must be selected with respect to the transformation T_l, however. The application of analogue complexing to a trading system is described next.

4. TRADING SYSTEMS

4.1 SYNTHESIS Of Generated Predictions In Modeling/Prediction Module

It is the goal of self-organizing modelling to generate models of optimal complexity in an objective way. However, there is a limited number of choices for each of the two following cases:

1. *parametric* models: the class of systems to be modeled (linear/non-linear), the maximum number of system dynamics (time lag) and a few process parameters; and
2. *non-parametric* models: the length of patterns, the variables considered, and the number of best patterns used.

Though a number of models with sufficient performance for a given data sample may exist, parametric and non-parametric models are only simplified reflections of economic reality. Each model is only an abstraction, a onesided reflection of some important behaviour of economic reality. As such, alternative parametric and non-parametric models can be used to estimate the vagueness of predictions.

By applying a number of prediction models, we may determine the need to decrease or increase the prediction interval based on the following guideline: the smaller the interval, the more certain a prediction should be. By observing

Signal Generation By Synthesis Of Predictive Information

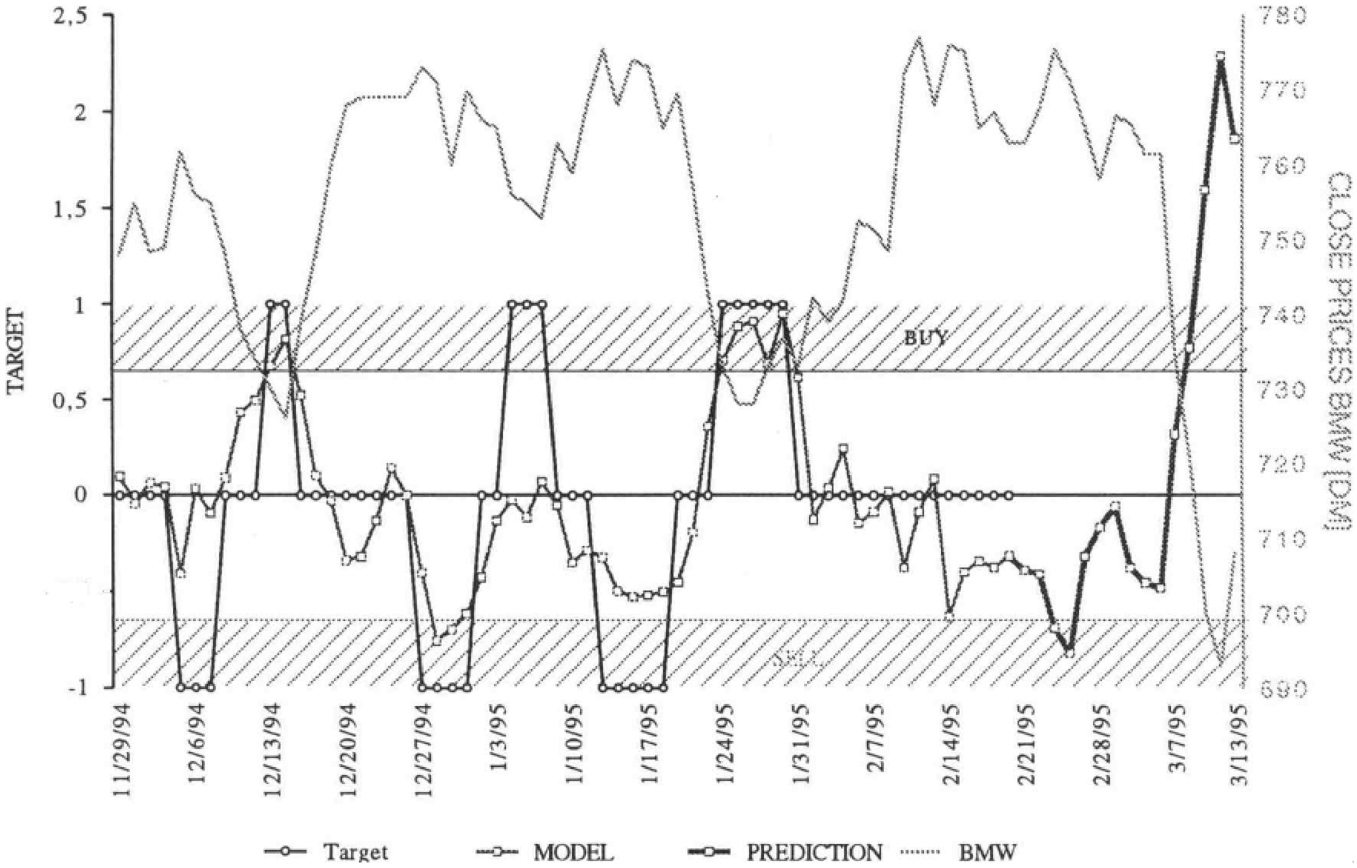


FIGURE 10. Synthesis of predictive information using **GMDH.**

a large number of such intervals over time, the most likely prediction of points is a combination (synthesis) of the predictions at a given point in time. In Figure 8, such an interval is shown along with its corresponding predictions.

4.2 EXTRACTION OF RULES (TRADING SIGNALS)

The second task in the development of our trading system is to use the information extracted by the modelling module (i.e., the predictions and their interval of uncertainty) to generate trading signals.

Using a wide spectrum of mathematical methods, a large number of trading indicators have been developed. Rather than creating a new indicator, we decided to consider modifying an existing one. We tested two alternative options:

1. A modified indicator based on the Moving Average Convergence Divergence (MACD) indicator. This indicator was chosen because of its relationship with our overall approach.
3. A classification system created by GMDH to produce buy, hold or sell states, using the outputs of the prediction module.

4.2.1 MODIFIED MACD INDICATOR

The principle of the MACD indicator is described as follows: when two, different, exponentially-smoothed curves (a shorter and a longer moving average of the same series) cross, a trading signal is generated (see Figure 9). As with other indicators computed using historical data, the MACD has one important disadvantage: since only historical data are used, trading signals will tend to lag the best trading points in time. This time delay may lead to significant losses.

Because of the time delay inherent in the MACD indicator, we were interested in minimizing this delay by simultaneously considering not only historical data but predictive information as well. For example, for the MACD, a 20-day moving average of a time series at time t is not exclusively computed on all past data beginning at y_{t-19}. Rather, given a forecast horizon of 7 days, it is computed on all data beginning at y_{t-12} up to y'_{t+7}, where y'_{t+n} represents a predicted value and y_{t-m} a historical value of the time series. This can be viewed as predicting the indicator 7 days ahead while shifting the MACD and signal line to the left. Predictive information are, in this case, daily out-of-sample predictions 7 days ahead. Their interval of uncertainty was calculated using several

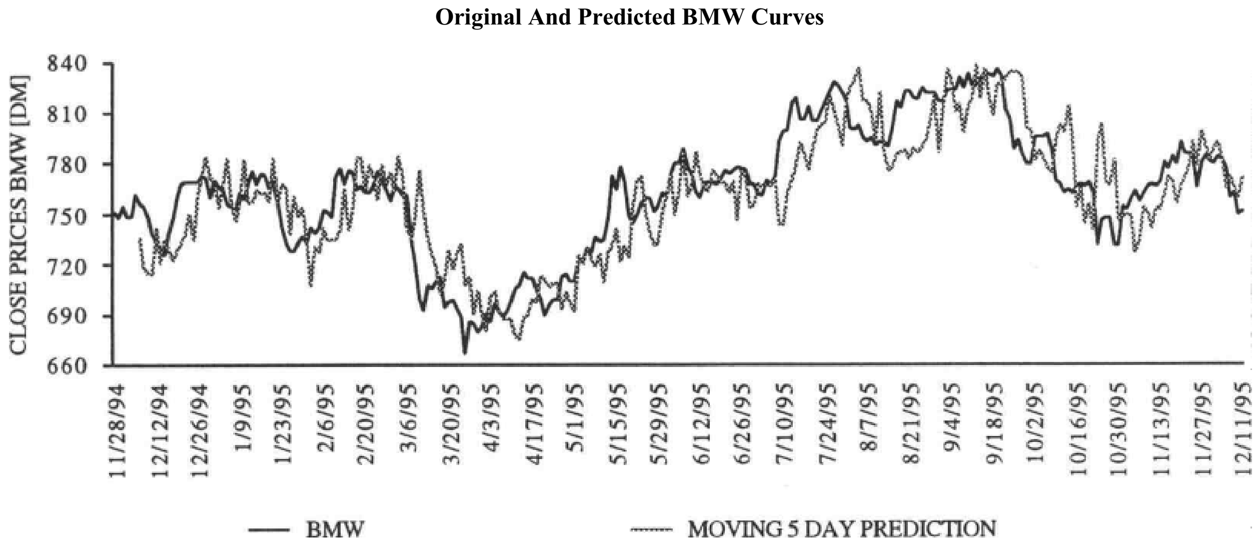


FIGURE 11. Moving 5-day out-of-sample predictions.

prediction models based on the methods described earlier.

Next, we applied a kind of worst case analysis with respect to the intervals of uncertainty. To do this, we considered the lower border of the prediction interval for evaluating whether or not a rising trend was expected and vice versa regarding falling trends. The modified MACD indicator (MODMACD) was then calculated for both cases based on the daily predicted asset curve along with its historical observations. As an alternative approach, one might consider calculating the indicator only once each time based on the synthesized predictions. Initial results indicate that this approach generates signals, on average, 3 days earlier than the unmodified MACD without the predictive information. Performance results for trading different stocks using this indicator are presented in Section 5.

4.2.2 SYNTHESIS OF PREDICTIVE INFORMATION

The previous method described for synthesizing predictive information is suitable for minimizing the time delay of indicators calculated using historical data. However, each indicator has its own strength and weaknesses depending upon the specifics of the application. Thus, it is necessary for one to know when to use a specific indicator in order to derive the most benefit from it. Because of this, we have developed another synthesis technique for generating a trading signal (SYNTHESIS). The idea was to compute not only the MODMACD indicator from the predictions but from a few additional variables as well. We describe these variables below:

- PRED — the most anticipated value of the **prediction;** that is, for example, the 5th value if the forecast horizon was 5,
- PSC — the most anticipated value of the corresponding **predicted smoothed curve,** which is a moving average calculated partly on predicted and partly on historical values,
- TREND — the ascent of the linear **trend** of the long-term prediction calculated using the predicted values exclusively, and
- MAD — the prediction error based on the **mean absolute deviation** calculated based on the forecast horizon *P* as soon as the corresponding true values were available. This implies storing *P* predictions for Pdays.

The above variables plus the **asset** curve itself (ASSET) provided the input variables to a GMDH network model for classification and signal generation. The **target** output variable (TARGET) was developed using a small training data set from the available time series as follows: a *buy*-signal was weighted by a 1, a *sell-signal* by a -1 and a *hold-signal* by zero. The only condition placed on the approach investigated was that the output variable must consist of at least two desired trades over the training data set.

Using these input and output variables, a GMDH model was trained to decide which combination of variables was best for predicting the behaviour of the output variable. For each new input vector, the trained model was used to classify the vector as a buy, sell or hold situation. For the BMW stock and ASSET = BMW, the following combination of variables was generated:

$$\text{TARGET} = 17.2363 + 0.1620\text{MAD} - 0.0330\text{BMW} + 0.0097\text{PRED}$$

A signal is generated if the absolute value of the TARGET is greater than the absolute value of a threshold, as follows:

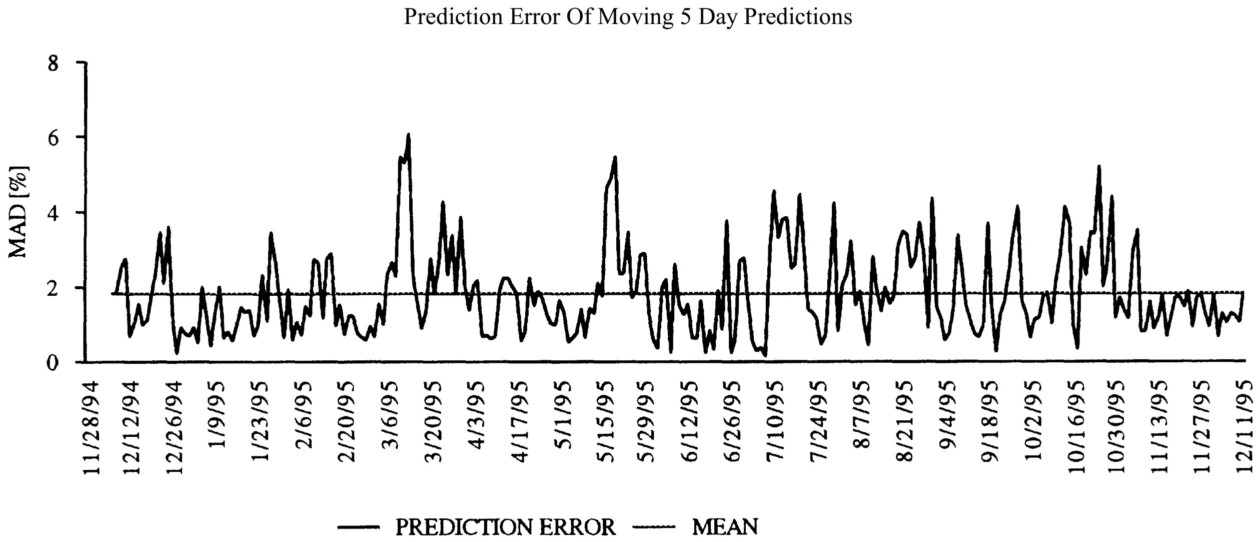


FIGURE 12. Prediction error (MAD [%]) of a predictor which uses analogue complexing.

BUY = TARGET > THRESHOLD, SELL = TARGET < -THRESHOLD.

As a threshold, we chose the value 0.6. This value was selected mainly as a result of our analysis of the model's ability to make robust classifications over a long time. Figure 10 graphically illustrates this synthesis process example over one time-period.

As is, the model appears to be able to detect significant trends changes. If updated daily, the adaptive behaviour of the model can be amplified by re-weighting the output variable as necessary in order to better reflect actual asset trends relative to previous trends.

5. PERFORMANCE OF THE TRADING SYSTEM

In our initial performance tests, we applied self-organizing data mining without considering the data selection steps mentioned earlier, since only a small number of time series samples were available.

We have tested 2 stocks of the German car industry (BMW and VW) as well as the US-Dollar/DM exchange rate over the period November 28, 1994 through December 11, 1995, and the S&P500 index over the period March 13, 1992 through December 30, 1994. All data considered were based on daily closing prices. Since our approach utilizes a systematic, daily adaptation approach to prediction modelling (due to the time variances of financial processes), the tests performed represent true, out-of-sample results over the test periods. All trades were computed without commission costs and were based on closing prices one day after a trading signal was generated. No stops or profit targets were used and no interest was earned.

The synthesis of different models on a daily basis is easily realizable. However, it is practically impossible to simulate this approach *ex post* over a long period of time without using a simulation/automation tool. Therefore, our performance tests were based exclusively on the predictions of different non-parametric models using analogue complexing techniques.

We began the modelling process by utilizing additional historical datasets to develop the first models. Next, we updated all of the initial models on a daily basis using each asset for the purpose of selecting the most representative market situations relative to the current situation. This is accomplished implicitly using analogue complexing techniques, since the process of comparing analogous and reference patterns results in either the manual or automatic selection of relevant market variables.

The final models were used to predict asset prices 7 days ahead and to generate the trading signals associated with the predictions. Figure 11 illustrates 5-day out-of-sample predictions along with the actual asset prices for BMW. Figure 12 provides a graph of corresponding prediction errors. Figures 13 and 14 illustrate performance results for BMW, VW and Dollar contracts using the three trading signals described earlier.

Table 4 lists the total returns for the BMW, VW, Dollar contracts and the portfolio using all 4 strategies.

CONCLUSIONS

Self-organizing data mining algorithms for a portfolio trading system were presented which realize a predictive control solution. To obtain predictions of financial markets appropriate for decision making, we have developed the following approach to modelling:

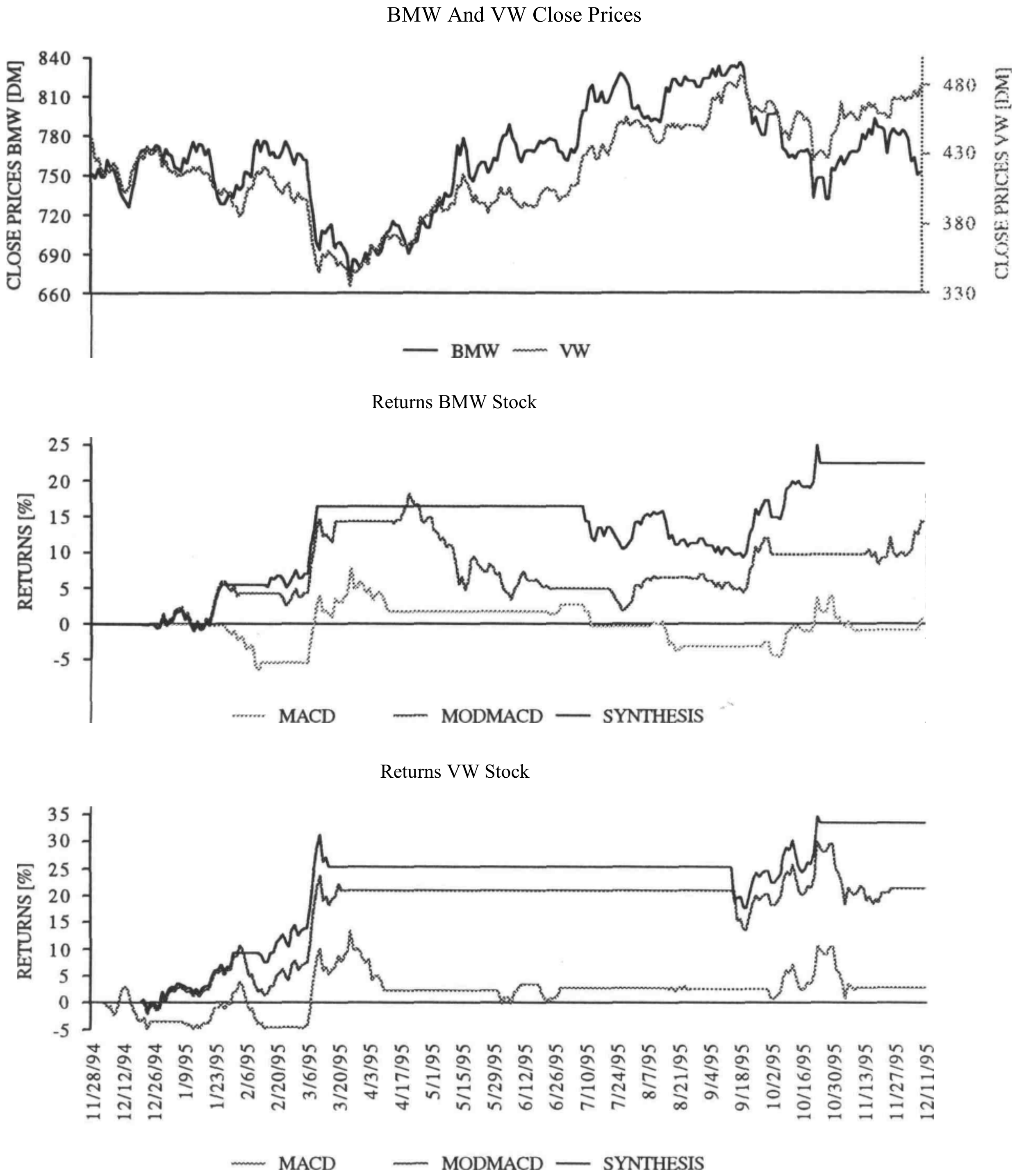


FIGURE 13. Performance results from the use of 3 different signal-generation methods.

	BM	W	VV	V	DOL	.LAR	PORTFOLIO
Trading Strategy	time in the market [%]	total return *m*	time in the market *[%]*	total return [%]	time in the market [%]	total return [%]	total 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
SYNTHESIS	52.40	22.55	66.05	41.27	58.67	8.04	71.86

TABLE 4. Total returns for different trading strategies.

- GMDH-type neural networks to create optimally complex, parametric regression models, which are analytically available by default (it was shown that the application of GMDH is also beneficial for efficient and systematic data reduction, synthesis and rule induction tasks),
- analogue complexing as a method for selecting similar market situations out of a given dataset of representative variables, and
- a synthesis of different models to reflect the vagueness of the future more appropriately.

A second task of the trading system was to transform the predictions made into trading signals. We tested two options in this regard:

- a modified MACD indicator and
- a synthesis of several types of predictive information using GMDH.

Initial performance results indicate that the prediction system presented appears to outperform a buy-and-hold strategy as well as an MACD-based trading system over the long-term for various assets. Since the MACD indicator was selected only as an initial example, further validation of results will include performance analysis using a larger spectrum of indicators, assets and markets.

Other directions of further research for improving the approach presented in this paper include:

- further perfection of algorithms for analogue complexing, especially the determination of appropriate pattern lengths and selection criteria,
- application of self-organizing fuzzy modelling techniques for improving the prediction models as well as the control module, and
- implementation of new (perhaps fuzzy) combining or synthesis methods.

REFERENCES

Cheng, W., Wagner, L. and C.-H. Lin [1996] "Forecasting the 30-year U.S. Treasury Bond with a System of Neural Networks," *NEUROVE$T JOURNAL,* Vol.4, No.1, pp. 10-16.

Goonatilake, S. and R. Feldman [1994] "Genetic Rule Induction for Financial Decision Making," in *Genetic Algorithms in Optimization, Simulation, Modelling (J.* Stender et al., eds.), IOS Press, Amsterdam, pp. 185-201.

Haasdijk, E.V. et al [1994] "Genetic Algorithms in Business," in *Genetic Algorithms in Optimization, Simulation, Modelling (J.* Stender et al., eds.), IOS Press, Amsterdam, pp. 157-

Dollar/DM Close Prices

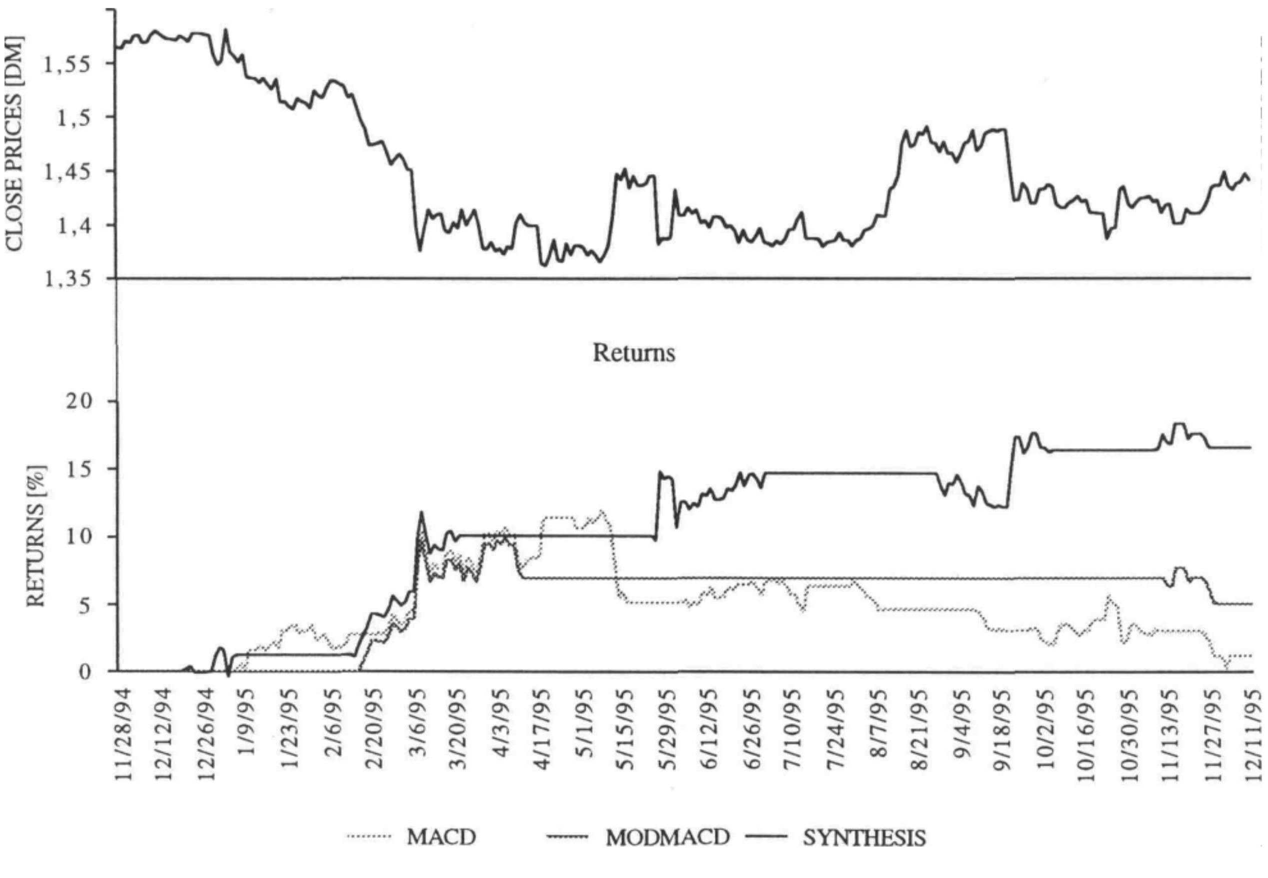


FIGURE 14. Performance results from trading the US-Dollar using 3 different signal-generation methods.

Ivakhnenko, A.G. and J.-A. Mueller [1992] "Parametric and non-parametric procedures in experimental system analysis," *Systems Analysis Modelling Simulation,* Vol.9 , No. 17, pp. 157-175.

Lemke, F. [1995] "SelfOrganize! - A Software Tool for Modelling and Prediction of Complex Systems," *Systems Analysis Modelling Simulation,* Vol.12, No. 20, pp.79-92.

Lorence, E.N. [1969] "Atmospheric predictability is revealed by naturally occurring analogues," *J. Atmos. Sci.,* Vol.4, pp. 636-646.

Madala, H.R. and A.G. Ivakhnenko [1994] *Inductive Learning Algorithms for Complex Systems Modelling,* CRC Press, Boca Raton, FL.

Wu, L. and S. Rehfuss [1997] "An Adaptive Committee-Based Approach to Trading," in *Non-linear Financial Forecasting: Proceedings of the First INFFC,* Finance & Technology Publishing, Haymarket, VA.

Frank Lemke *is an independent software developer specialized in self-organising modelling based solutions and information systems for over 5 years. He has developed SelfOrganize!, a tool which uses an advanced GMDH algorithm as well as the analogue complexing method for the modelling and prediction of complex systems. He graduated from the Institute of Electrical Engineering at Humboldt-University in Berlin in 1993. Since 1992 he worked on several problems in economics and finance such as balance sheet prediction, sales prediction, solvency checking, and predictive control solutions for financial markets.*

Johann-Adolf Mueller *graduated from the Ilmenau Institute of Electrical Engineering in 1965. He obtained his Ph.D. in Economics from the Institute of Economics in Berlin in 1978, with a focus on dynamic modelling in economics. Since 1992, Prof. Mueller has been working at the Hochschule fuer Technik und Wirtschaft in Dresden in the field of business-oriented computer science.*