Forecasting Commodity Prices for Predictive Decision Support Systems

Tamer Shahwan [a], Frank Lemke [b]

[a] *Humboldt-Universität zu Berlin, School of Business and Economics, Institute of Banking, Stock Exchanges and Insurance, D-10178 Berlin, Germany and Faculty of Agriculture and Horticulture, Department of Agricultural Economics and Social Sciences D-10099 Berlin, Germany; tamer.shahwan@agrar.hu-berlin.de*

[b] *KnowledgeMiner Software, D-16341 Panketal, Germany, frank@knowledgeminer.net*

Abstract

The builders of traditional decision support systems have regularly used game theory and operations research to build intelligent decision support systems. However, we have seen a rapid acceptance of new technology like neural networks and data mining which implement state-of-the art decision support system based on knowledge management to solve a wide range of business problems like classification and forecasting. Thus, we will illustrate our claims through the investigation of the forecasting performance of two time series forecasting techniques, namely elman neural network and self-organizing data mining against the autoregressive integrated moving average (ARIMA) model and futures prices as benchmarks. As an attempt to improve the accuracy of elman neural networks, the global search capability of Genetic Algorithms will be used to determine the optimal architecture of elman neural network. Real data sets of commodity prices are used to examine the forecasting accuracy of the proposed models.

Key words: Decision support systems (DSS); Time Series Forecasting; ARIMA; Elman Neural network; Self-organizing Data Mining (SODM).

1 Introduction

Since the early 1970s, decision support systems (DSSs) technology and its applications have evolved significantly. According to Turban (1993) a decision support system couples the intellectual resources of individuals with the capabilities of the computer to improve the quality of decisions and to deal with semi-structured problems. Turban and Aronson (1998) extended the definition of DSS as a computer based information system that combines models and data in an attempt to solve non-structured problem with extensive user involvement.

At the early beginning of 1990s, modern decision support systems research is focused on the theory and applications of intelligent systems and soft computing in management. Forecasting is one of the most popular fields in decision suport system applications, in particular using data mining methods such as artificial neural networks (ANNs) and fuzzy neural networks (Kuo and Xue, 1998). Zhang et al. (1998) summarize the different applications of artificial neural networks for forecasting. Kaastra and Boyd (1996) provide a general introduction of how neural network model should be developed to model financial and economic time series. Mueller and Lemke (2000) describe the theory and application of advanced and unique self-organizing data mining technologies like inductive, Statistical Learning Networks to various problem fields in economy (modelling and prediction of national economies, sales prediction), finance (price prediction and financial trading), or ecology (water and air pollution problems, drainage outflow).

The popularity of ANNs as a generalized non-linear forecasting model is derived from several distinguishing features that make them a valuable and attractive tool for this purpose. First, in comparison to traditional model-based methods, ANNs are data-driven methods. They require few a priori assumptions about the models for problems under study (Zhang et al. 1998). Second, based upon the presentation of several input patterns and their associated outputs, ANN is able to autonomously learn the map from inputs to output (Richards et al. 1998). Hence, ANNs are more appropriate for complex phenomena, especially when we have a poor understanding of the relationships within these phenomena (Gorr et al. 1994). Finally, although economic theory can suggest the elements of the input and output set, the complexity of the real system and significant non-linearity in the relationship introduced through the inclusion of risk preferences, and imperfect competition may suggest that ANN can effectively fit the data better than do traditional statistical forecasting methods (Richards et al. 1998).

Our motivation for predicting agricultural commodity prices in the current study is: First, there are few articles using artificial neural networks for predicting agricultural commodity prices. For instance, Kohzadi et al. (1996) use backpropagation networks to model monthly live cattle and wheat prices and compare the result with that obtained by the autoregressive integrated moving average (ARIMA) model. They conclude that neural networks perform better in terms of mean absolute error (MAE) and mean absolute percentage error (MAPE). Moreover, neural networks were also better in capturing turning points. Richards et al. (1998) compare two methods of estimating a reduced form model of fresh tomato marketing margins: an econometric and an artificial neural network approach. The neural network is able to forecast with approximately half and mean square error of the econometric model, but both are equally adept at predicting turning points in the time series. Second, agricultural commodity prices are more volatile than are prices of most non-farm goods and services (Tomek and Robinson, 2003). Thus, getting sufficiently accurate commodity prices forecasts will be more helpful for price risk management to deal with random deviation of cash prices around the known systemic pattern. Third, obtaining an effective and accurate price forecasting will support decision maker towards a variety of decisions, such as production decision, storage decision, hold and sell decisions and hedging decision. Finally, there is justifiable scepticism in the ability to make money by predicting prices changes in any given market. This scepticism reflects the efficient market hypothesis according to which markets fully integrate all of the available information in the prices and prices fully adjust immediately once new information becomes available. Consequently, we aim to investigate the ability of these forecasting methods against the efficient market hypothesis.

We examine the use of elman neural network (ENN), evolutionary elman neural network (EENN)[1] and self-organizing data mining (SODM) represented by the Group Method of Data Handling (GMDH) algorithm for forecasting monthly hog prices and monthly rape prices in German Market. This study turns out that the suitable model should be embedded in the model base of a decision support system. In terms of model evaluation, both ARIMA and futures prices are used as benchmarks. The usefulness of each method to predict monthly prices will be assessed by means of an out-of-sample technique.

The rest of the article is organised as follows: section 2 briefly describes the forecasting methods like ENN, GMDH, ARIMA and futures prices while section 3 describes the empirical analysis and the experimental design of forecasting models. Section 4 provides the results of the comparison followed by final remarks and conclusions in section 5.

2 Time series forecasting models

2.1 Elman Neural Network

The elman neural network (ENN) is one kind of globally feedforward locally recurrent network model proposed by Elman (1990). ENN can be considered as an extension of multilayer perceptron (MLP) with an additional input layer (state layer) that receives a feedback copy of the activations from the hidden layer at the previous time step. These context units in the state layer in the Elman network make it sensitive to the history of input data, which is essentially useful in dynamical system modelling (Yang et al. 2004).

[1] Evolutionary elman network implies using genetic algorithm with ENN to optimize the following parameters (hidden nodes, learning rate value, momentum rate value and the value of weight - decay constant λ).

Fig. 1 depicts the idea of ENN where the activations in the hidden layer at time *t*-1 are copied into the context vector as input to the network at time *t* (Cho, 2003).

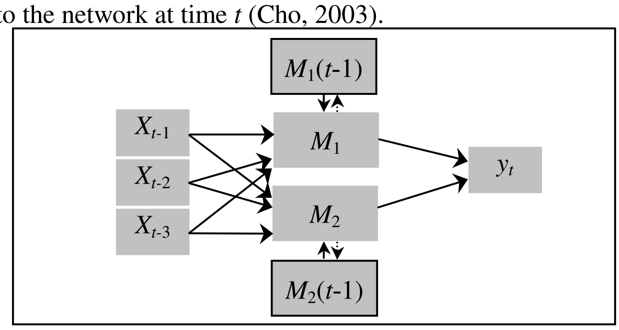

Fig.1 Elman neural network (McNelis 2005)

An ENN with a "tanh sigmoidal activation function" has the following structure (McNelis 2005):

$$n_{k,t} = w_{k,o} + \sum_{i=1}^{i*} w_{k,i} x_{i,t} \tag{1}$$

$$M_{k,t} = \left(e^{n_{k,t}} - e^{-n_{k,t}}\right) / \left(e^{n_{k,t}} + e^{-n_{k,t}}\right) + \sum_{k=1}^{k*} \phi_k M_{k,t-1} \tag{2}$$

$$y_t = \gamma_0 + \sum_{k=1}^{k*} \gamma_k M_{k,t} \tag{3}$$

Where x_i are input variables (1,2,……,*i**). *k** is the number of neurons, $w_{k,0}$ is a constant term, $w_{k,i}$ are the synaptic weights of input variables, $n_{k,t}$ is a linear combination of these input variables observed at time *t*, γ_0 is the constant term, γ_k is the coefficient between the hidden and the output layer and ϕ_k are weights between context units and hidden units. Hence, the output (y_t) depends not only on the new inputs but also on the preceding context units.

2.2 Group Method of Data Handling

The Group Method of Data Handling algorithm (GMDH) was developed by A.G. Ivakhnenko in 1967, and considerable improvements were introduced in the 1970s and 1980s by versions of the Polynomial Network Training algorithm (PNETTR) by Barron and the Algorithm for Synthesis of Polynomial Networks (ASPN) by Elder. Barron and Barron (1988) give an excellent overview about such early network developments.

Experience has shown that there are several problems in the development and application of neural networks as follows (Mueller and Lemke, 2000): (1) Neural network models are commonly implicit models that have an internal connective representation of knowledge. No explanation component (analytical model) is provided by default (black box problem). (2) designing a neural networks' topology is a trial-and-error process. (3) there are no rules how to use the theoretical a priori knowledge for ANN design. (4) Neural network learning techniques are a kind of statistical estimation often using algorithms that are slower and less efficient than highly optimised algorithms used in statistical software. (5) The theory of neural networks does not introduce a general principle for avoiding the overfitting problem. If neural networks are created on noisy data samples, which always includes data sets of few samples, ANN will usually overfit the design data and will generalize poorly.

Consequently, several new or modified techniques including methods for avoiding overtraining like the cross-validation technique have been presented and implemented in various software tools to overcome some of these problems related to neural network design. Another approach, Statistical Learning Networks (SLN), which has been developed in parallel to Neural Networks, can solve systematically the first three problems listed. Extending SLNs by an inductive approach leads to self-organising modelling in form of GMDH algorithms as a most used representative of SLNs. GMDH also intend to solve the problems 4 and 5 by providing a sound scientific foundation (Mueller and Lemke, 2000). This makes GMDH a most automated, fast and very efficient supplement and alternative to other data mining methods. Also, in result of modelling an analytical model in form of algebraic formulas, difference equations, or systems of equations is available on the fly for interpretation and for gaining insight into the system.

The GMDH algorithm objectively selects the model of optimal complexity using an inductive approach. This includes the use of external information that was not used for estimating the coefficients of the model.

The GMDH utilized here generates an optimised 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 nonlinear polynomials. The general relationship between input and output variables can be described in form of a functional volterra series, whose discrete analogue is known as the Kolmogorov-Gabor polynomial (Ivakhenenko and Ivakhenenko, 1995),

$$y_t = a_0 + \sum_{i=1}^{M} a_i x_i + \sum_{i=1}^{M}\sum_{j=1}^{M} a_{ij} x_i x_j + \sum_{i=1}^{M}\sum_{j=1}^{M}\sum_{k=1}^{M} a_{ijk} x_i x_j x_k \qquad (4)$$

where $X(x_1, x_2, ..., x_M)$ is the vector of input variables and $A(a_1, a_2, ..., a_M)$ is the vector of the summand coefficients. Then, the problem is to estimate the polynomial degree (model complexity) and its parameters. The original GMDH algorithm basically consists of the following three principles (Mueller and Lemke, 2000): a) the cybernetic principle of self-organization as an adaptive creation of a network without subjective points given, b) the principle of external complement enabling an objective selection of a model of optimal complexity where the external criterion is computed from new information that was not used for estimating the coefficients of the model and c) the principle of regularization of ill-posed tasks.

2.3 Box-Jenkins ARIMA model

The familiar Box-Jenkins approach combines two types of processes: autoregressive (AR) and moving average (MA). The general class of ARMA (*p,q*) model has the following form (Tsay, 2002):

$$(1 - \phi_1 B - \phi_p B^p) y_t = \phi_0 + \left(1 - \theta_1 B - ..... - \theta_q B^q\right) a_t \qquad (5)$$

where $\{a_t\}$is a white noise series, *p* and *q* are non negative integers, B is the back shift operator with $B(y_t) = y_{t-1}$ where y_t denotes the monthly prices. This model to be meaningful, we need $\phi_1 \neq \theta_1$; otherwise there is a cancellation in the equation and the process reduces to a white noise series. Since many time series are nonstationary, differencing one or more times is required. This leads to an ARIMA *(p,d,q)* model. (Pindyck and Rubinfeld, 1991):

$$(1 - \phi_1 B - \phi_2 B^2 - .... - \phi_p B^p)(1 - B)^d y_t = \left(1 - \theta_1 B - \theta_2 B^2 - .... - \theta_q B^q\right) a_t \qquad (6)$$

Furthermore, the regular and seasonal components of a time series can be captured by a general multiplicative ARIMA model (Nelson, 1973):

$$\phi_p(B)\Phi_P\left(B^s\right)(1 - B)^d\left(1 - B^s\right)^D y_t = \theta_q(B)\Theta_Q(B^S)a_t \qquad (7)$$

where B^s is the seasonal backshift operator, $\Phi_p(B^s) = (1 - \phi_{1s}B^s - \phi_{2s}B^{2s} - ...... - \phi_{ps}B^{ps})$ is the seasonal autoregressive process and $\Theta_Q(B^s) = (1 - \Theta_s B^s - \Theta_{2s} B^{2s} - .... - \Theta_{Qs} B^{Qs})$ is the seasonal moving average process. In general, *s* equals 4 or 12. *P*, *D*, *Q* have values of 0, 1 or 2. A useful notation to describe the orders of the various components in this multiplicative model is given by $(p, d, q) \times (P, D, Q)^s$ where *P* is the seasonal level of auto-regressions, *D* is the seasonal level of differences and *Q* is the seasonal level of moving average. Basically, the application of ARIMA models consists of three phases: model identification, parameter estimation and diagnostic checking. The identification step requires an intensive data analysis where expert judgment must be exercised to interpret the behaviour of the autocorrelation function (ACF) and the partial autocorrelation function (PACF). However, any significant nonlinearity limits the application of ARIMA models. Therefore, elman neural network and SODM are proposed to deal with a non-linear pattern possibly present in the data.

2.4 Futures Prices

Recently, futures markets enable noumerous companies, ranging from small to large business, to better manage their price risk. Kolb (1997) summarizes two main social functions of futures markets as follows: (1) price discovery where the market observers can form estimates of what a price of a given commodity will be at a certain time in the future by using the information contained in futures prices today. (2) Hedging where many futures market participants trade futures as a substitute for a cash market transaction to reduce a pre-existing risk. Since our commodities under investigation are mainly traded in Commodity Exchange Hannover/WTB, then futures prices of those commodities can serve as a useful benchmark and price discovery mechanism. Like most futures contract, a contract is often open for many months, and all of the subsequent daily futures prices reflect the changing market expectations of what the spot price will be on the last day of trading (Tomek and Robinson, 2003). Accordingly, the futures prices for one month ahead forecasting is selected by working backward 28-days from the contract termination date of the target month (Kellard et al. 1999). These forecasts will be compared to the forecasts made by elman neural network and SODM.

3 Empirical analysis

3.1 Data set

Two data sets, monthly hog prices and rape prices from Germany, are now used to investigate the effectiveness of forecasting conducted by previous mentioned models[2]. We assess the forecasting performance by means of an out-of-sample-technique. Each time series is divided into a training set and testing set. The training set is used for model specification and then the test set is used to evaluate the established model. In order to achieve a good generalization of the ENN, a cross-validation approach is adopted. That means the training set is further portitioned into two subsets: 80 percent of the training set is assigned to the estimation and the remaining 20 percent is assigned to the validation subset (Haykin, 1999). Before estimation the data are normalized to the range between [-1, 1]. The data decompositions of the two data sets are given in table 1.

The measurement of prediction performance of the four models is based on the effectiveness which refers to the prediction accuracy of the model based on one-step-ahead forecasts over the test set. The mean absolute percentage error (MAPE) and Theil's are employed to measure the forecasting errors.

$$MAPE = 100/T \sum \left|(\hat{y}_t - y_t)/y_t\right| \tag{8}$$

$$Theil's\,U = \left(\frac{1}{T}\sum_{t=1}^{T} (\hat{y}_t - y_t)^2\right)^{1/2} \Bigg/ \left(\left(\frac{1}{T}\sum_{t=1}^{T}(\hat{y}_t)^2\right)^{1/2} + \left(\frac{1}{T}\sum_{t=1}^{T}(y_t)^2\right)^{1/2}\right) \tag{9}$$

where y_t and $\hat{y}_t$ are the actual and the predicted price, respectively, at time t and T is the number observations in the test set.

[2] The data are obtained from ZMP "Zentrale Markt- und Preisberichtstelle GmbH- Marktberichtstelle" (ZMP), Berlin.

Table 1 Sample decomposition (in parenthesis: absolute number of samples)

Series	Total Sample	Training set	Cross-validation set	Test set
Hog prices (€/kg slaughter weight)	1972-2003 (384)	1972-1999 (336)	20% of training set (67)	2000-2003 (48)
Rape prices[3] (€/ton)	1992- July 2004 (151)	1992-2001 (120)	20% of training set (24)	2002-july 2004 (31)

3.2 Specification of the forecasting models

The estimated ARIMA model for the hog data has the structure $(2,0,0)x(1,0,1)_{12}$. The rape data is fitted best with an autoregressive model (AR) of order 3. Next, we test for the presence of nonlinearities in the data. If nonlinearities are statistically significant, then choosing a class of nonlinear models like ANN might improve the predictive power. In this context, we apply the Ljung-Box-Q-statistic to the squared residuals of the two ARIMA models.

The results in table 2 show that there is evidence for the presence of GARCH effects (i.e. conditional heteroscedasticity) in the hog prices. No such effect occurs in the rape data. Conditional heteroskedasticity implies that the underlying times series is nonlinear in variance. Baillie and Bollerslev (1992) show that conditional heteroscedasticity changes the mean square error (MSE) of the predictor. Yim (2002) successfully applies ANN to data that show conditional heteroscedasticity. In his study ANN prove to be superior to ARIMA-GARCH models. Due to this finding we conjecture that ENN will outperform the traditional ARIMA models and futures prices particularly for the hog price data.

The implementation of the ENNs requires to specify a large number of parameters. In order to specify the ENNs, we apply a mixture of different methods. First, the number of neurons, m, in the input layer is heuristically determined by the number of autoregressive (AR) terms of the ARIMA models. Hence, the variables (y_{t-1}, y_{t-2}, y_{t-12}, y_{t-13}, y_{t-14}) are used as inputs to the ENN in the case of the hog prices. The corresponding input variables for the rape prices are (y_{t-1}, y_{t-2}, y_{t-3}). The hyperbolic tangent function is chosen as a transfer function between the input and the hidden layer. The identity transfer function connects the hidden and the output layer. Second, the weight decay method as pruning technique is used to reduce the weights and to find a parsimonious structure for the ENN.

In the final hog price model, the initial settings of learning parameters are as follows: the learning rate is 0.9, while the momentum term is 0.7 for the hidden layer but in the output layer the learning rate is 0.1, while the momentum term is 0.7. The best forecast accuracy is obtained when the initial weight values are between [-0.5 and 0.5]. The ENN is trained by backpropagation. Batch updating is chosen as the sequence, in which the patterns are presented to the network. As mentioned above the cross-validation approach is used to determine the optimal number of training epochs. That means the training procedure is terminated as soon as the mean square error of the cross-validation set increases, since this indicates that the network has begun to overtrain. This prevents the ENNs from memorizing unnecessary noise in time series.

In a similar fashion, the initial settings of learning parameters for rape price model are as follows: the learning rate is 0.9, while the momentum term is o.7 for the hidden layer but in the output layer the learning rate is 0.1, while the momentum term is 0.7. The number of hidden nodes was set to 9 nodes after experiments with trial and error. The hyperbolic tangent function is chosen as transfer function between the input and the hidden layer. The identity transfer function connects the hidden and the output layer.

[3] Monthly rape prices include missing data in the following months: July 92,June 93, April 94, Mai 94, June 94, July 94,July 95,July 96, July 97, July 98, July 2000, July 2002 and July 2003. The interpolation method is used for replacing the missing observations.

Table 2 The Ljung-Box statistics for the squared residuals of an ARIMA model

Time series	Q (5)	Q (10)	Q (15)	Q (20)	Q (24)
Hog Prices	36.27	43.97*	51.30*	81.18*	89.53*
Rape Prices	6.10	6.11	6.12	6.12	6.13

* Denotes statistical significance at the 5% level

Since the error surface implies several characteristics (i.e. infinite large, non-differentiable, complex, noisy, deceptive and multimodal), then the GA-based evolutionary approach is a better candidate for searching the surface than the heuristic approach like constructive/destructive algorithms (Yao, 1993). Hence, genetic algorithms (GAs) as global search procedure are used to support configuration process of the ENN. In particular, the number of hidden nodes, the value of learning and momentum term and a weight-decay constant λ.

For creating linear and nonlinear models by GMDH, we have chosen a time lag of up to 37 months as the set of potential input variables. By defining just two parameters, number of samples and number of potential input variables (number of time lags in the case of autoregressive models), it is possible to self-organize an optimal complex model from data, which is composed then of a small set of self-selected relevant inputs. For every model created, information about its expected validity and an analytical model in form of a regression equation is provided on the fly. In the final hog price model, the nonlinear model created by GMDH is :

$$Y_t = 0.328- 0.205\, Y_{t-2} + 0.124\, Y_{t-9} - 0.0735\, Y_{t-13} - 0.1204\, Y_{t-4} + 1.154\, Y_{t-1} + 0.0227\, Y_{t-2}\, Y_{t-3} - 0.0691\, (Y_{t-2})^2 \tag{10}$$

In a similar fashion, the same approach was chosen for rape data and the self-organised regression model created by GMDH is:

$$Y_t = 25.49 - 0.01554\, Y_{t-25} + 1.438\, Y_{t-1} - 0.5152\, Y_{t-2} - 0.05999\, Y_{t-13} + 0.02531\, Y_{t-7} \tag{11}$$

4 Results

In the current study, all ARIMA models are estimated using Statgraphics Plus software. The software package NeuroSolutions V4.32 (NeuroDimensions Inc., Gainesville, FL) was employed for the estimation of the Elman Network. NeuroSulutions includes also genetic optimization to optimize the structure of ANN. The software package KnowledgeMiner (KnowledgeMiner, 2004) was employed to create linear and nonlinear polynomial model by using GMDH algorithm. The out-of-sample forecast performance of various models for both hog and rape data are reported in table 3 (see also Fig. 2 and Fig. 3).

The figures indicate that only slightly differences in the predictive power of the various methods exist. With regard to the hog price data, the combination of genetic algorithms and elman neural network (EENN) leads to a reduction of the overall forecasting errors in terms of MAPE compared to other forecasting models. While the MAPE of EENN is smaller than ENN the opposite is true for Theil’s U. With regard to the rape price data, no improvement of the forecasting accuracy can be achieved by using EENN. This raises some doubt, if the optimal specification of the ENN has been identified for this time series. The use of GMDH leads to slightly improvements in terms of MAPE and Theil’U relative to other forecasting methods. In accordance with our initial guess the EENN outperforms other forecasting models for the hog prices which obey a more complex and non-linear pattern. It should be mentioned that the differences in the prediction errors are not statistically significant at 5% level based on a t-test.

Table 3 Comparison of forecasting indices

Time series	MAPE (Rank)	Theil's U (Rank)
ENN		
(1) Hog prices	4.6296 (3)	**0.028858 (1)**
(2) Rape prices	3.4393 (4)	0.022895 (4)
EENN		
(1) Hog prices	**4.4741 (1)**	0.029691 (2)
(2) Rape prices	3.2333 (3)	0.022469 (3)
GMDH		
(1) Hog prices	5.0934 (4)	0.031944 (4)
(2) Rape prices	**2.9979 (1)**	0.0**21680 (1)**
ARIMA		
(1) Hog prices	4.5463 (2)	0.030431 (3)
(2) Rape prices	3.0841 (2)	0.022246 (2)
Futures Prices		
(1) Hog prices	5.8073 (5)	0.034739 (5)

Bold letters indicate minimal errors

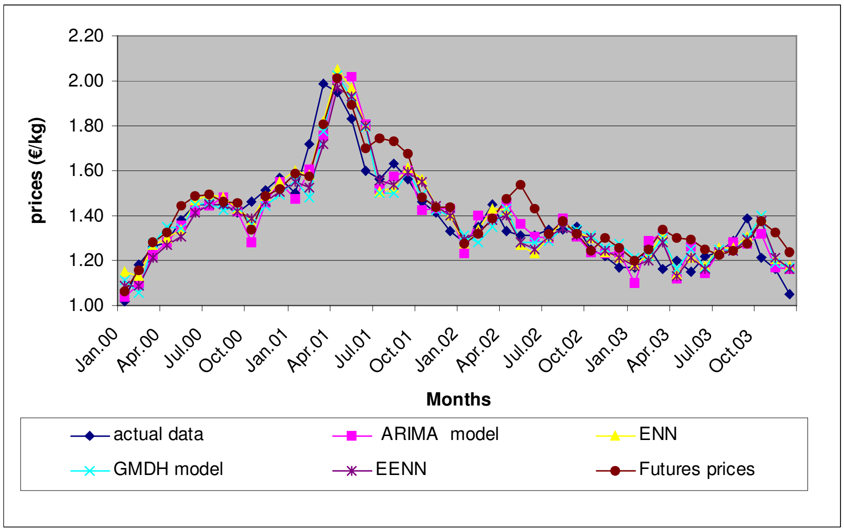

Fig. 2 one-step-ahead forecasts of hog prices (Germany, January 2000 to December 2003)

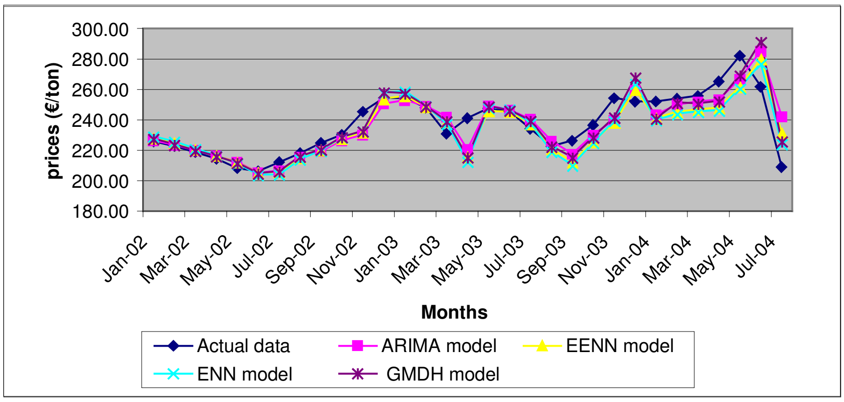

Fig. 3 one-step- ahead forecasts of rape prices (Germany , January 2002 to July 2004)

5 Conclusions

In this study we explored the usefulness and applicability aspects of ANN and Self-Organising Data Mining for short-term forecasting of agricultural commodity prices. Traditional ARIMA models and futures prices served as benchmarks for prediction performance evaluation. We also investigated whether a combination between elman neural network and genetic algorithms generates more predictive model compared to the other forecasting methods. Obviously, the potential gain of ANN,ARIMA and SODM seems to depend on the characteristics of the time series under consideration. A more complex time series justifies the use of ANN. However, the greater flexibility of this model class and its ability to handle nonlinear data patterns comes at the cost of a more demanding specification procedure and computation time, especially when applying GA for optimisation purpose, while one advantage of SODM from an application point of view is that it takes by far least efforts both in manual (almost nothing) and in computational time. Furthermore, it provides on the fly after each modelling run a reliable, analytical model that describes and does not overfit the design data. On the other hand, ARIMA needs a lot of manual efforts, theoretical knowledge and experience in determining the suitable model but it is straight forward in computation.

In contrast to ARIMA and SODM, there is no simple clear-cut method or theory on hand for determining the optimal structure of the ANN. Creating a reliable neural network model is trial-and-error process. This means the danger of misspecifying an ANN is higher than for an ARIMA model and SODM. This may erode the potential predictive power of ANNs even if our calculations confirm that application of GA for optimising the network topology of the ANN can slightly improve its predictive performance.

For further research, we expect that the results of this study can be improved by model combining and by adding additional information like macroeconomic characteristics or price data of other commodities and then building multi-input models from this extended data base.

6 References

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

Baillie, R.T., Bollerslev, T., 1992. Prediction in dynamic models with time- dependent Conditional Variance. Journal of Econometrics 52, 91-113.

Cho, V., 2003. A comparison of three different approaches to tourist arrival forecasting. Tourism Management 24, 323-330.

Elman, J.L., 1990. Finding structure in time .Cognitive Science, 14, 179-211.

Gorr, W.L., Nagin, D., Szczypula, J. 1994. Comparative study of artificial neural network and statistical models for predicting student grade point average. International Journal of Forecasting 10, pp. 17-34.

Haykin, S., 1999. Neural Network: A Comprehensive Foundation. Second Edition, Prentice-Hall Inc.

Ivakhnenko, A.G., Ivakhnenko, G. A., 1995. The review of problems solvable by algorithms of the group method of data handling (GMDH). Pattern Recognition and Image Analysis 5, 527-535.

Kaastra, I., Boyd, M., 1996. Designing a neural network for forecasting financial economic time series. Neurocomputing 10, 215-236.

Kellard, N., Newbold, P., Rayner, T., Ennew, C., 1999. The relative efficiency of commodity futures markets. The Journal of Futures Markets 19, 413-432.

Kohzadi, N., Boyd, M.S., Kermanshahi, B. 1996. A comparison of artificial neural network and time series models for forecasting commodity prices. Neurocomputing 10, 169-181.

Kolb, R. W., 1997. Understanding Futures Markets. Fifth Edition, Blackwell Publishers Ltd.

Kuo, R.J., Xue, K.C., 1998. A decision support system for sales forecasting through fuzzy neural networks with asymmetric fuzzy weights. Decision Support Systems 24, 105-126.

Knowledgeminer: Self-organising data mining and prediction tool. Version X 5.0.8 http://www.knowledgeminer.com

McNelis, P.D., 2005. Neural Networks in Finance: Gaining predictive Edge in the Market. Elsevier Academic Press.

Mueller, J.-A., Lemke, F., 2000. Self-Organising Data Mining: Extracting Knowledge from Data. BoD, Hamburg.

Nelson, C.R., 1973. Applied Time Series Analysis for Managerial Forecasting: Holland-Day, Inc., San Francisco.

Richards, T.J., Patterson, P.M., Ispelen, P.V., 1998. Modelling fresh tomato marketing margins: econometrics and neural networks. Agricultural and Resource Economics Review 27,186-199.

Turban, E., 1993. Decision Support and Expert Systems: Management Support Systems. Macmillan, New York.

Turban, E., Aronson, J., 1998. Decision Support Systems and Intelligent Systems. Prentice-Hall, London.

Tomek, W.G., Robinson, K.L. 2003. Agricultural Product Prices. Cornell University Press.

Tsay, R.S., 2002. Analysis of Financial Time Series. John Wiley & Sons, Inc.

Yao, X., 1993. A review of evolutionary artificial neural networks. International Journal of Intelligent Systems 8, 539-567.

Pindyck, R.S., Rubinfeld, D. L., 1991. Econometric Models and Economic Forecasts. McGraw-Hill, Inc.

Yang, F., Sun, H., Tao, Y., Ran, B., 2004. Temporal difference learning with recurrent neural network in multi-step ahead freeway speed prediction. TRB 2004 Annual Meeting.

Yim, J., 2002. A comparison of neural networks with time series models for forecasting returns on a stock market index. Proceedings of the 15th International Conference on Industrial and Engineering. Applications of Artificial Intelligence and Expert Systems (IEA/AIE), Cairns, Australia.

Zhang, G., Patuwo, B.E., Hu, M.Y., 1998. Forecasting with artificial neural networks: the state of the art. International Journal of Forecasting 14, 35-6