HYBRID FUZZY NEURAL NETWORK TO PREDICT PRICE DIRECTION IN THE GERMAN DAX-30 INDEX

Fernando GARCÍA[1*], Francisco GUIJARRO[2], Javier OLIVER[1],
Rima TAMOŠIŪNIENĖ[3]

[1]*Department of Economics and Social Sciences, Faculty of Business Administration,
Universitat Politècnica de València, Cami de Vera, s/n, 46022, València, Spain*
[2]*Insititut de Matemàtica Pura i Aplicada, Universitat Politècnica de València,
Cami de Vera, s/n, 46022, València, Spain*
[3]*Department of Financial Engineering, Faculty of Business Management,
Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223, Vilnius, Lithuania*

Abstract. Intraday trading rules require accurate information about the future short term market evolution. For that reason, next-day market trend prediction has attracted the attention of both academics and practitioners. This interest has increased in recent years, as different methodologies have been applied to this end. Usually, machine learning techniques are used such as artificial neural networks, support vector machines and decision trees. The input variables of most of the studies are traditional technical indicators which are used by professional traders to implement investment strategies. We analyse if these indicators have predictive power on the German DAX-30 stock index by applying a hybrid fuzzy neural network to predict the one-day ahead direction of index. We implement different models depending on whether all the indicators and oscillators are used as inputs, or if a linear combination of them obtained through a factor analysis is used instead. In order to guarantee for the robustness of the results, we train and apply the HyFIS models on randomly selected subsamples 10,000 times. The results show that the reduction of the dimension through the factorial analysis generates more profitable and less risky strategies.

Keywords: Trend forecasting, stock exchange index, technical indicators, artificial neural networks, fuzzy rule-based systems, HyFIS.

JEL Classification: C45, G17, G23.

Introduction

The efficient-market hypothesis was first developed by Fama (1970). He suggested three different types of market efficiency degrees depending on the information available and its impact on asset prices, i.e., whether the information was reflected or not in price evolution.

*Corresponding author. E-mail: fergarga@esp.upv.es

If markets are efficient, all information is already captured by the prices and it is not possible to beat the market.

The success of speculative investment strategies in the financial markets depends on market efficiency. If investors believe that the market is efficient, then they will choose passive investment strategies applying index tracking techniques (García, Guijarro, & Moya, 2013). These strategies imply following the evolution of a market index such as DJIA, Nikkei-225, DAX-30, etc. (García, Guijarro, & Moya, 2011; Guijarro, 2017).

If investors do not believe in market efficiency, they will prefer active strategies based on fundamental analysis, technical analysis or both of them. Albeit it still remains a controversial issue (Lim & Brooks, 2011), many research papers have reported some inefficiencies for distinct financial markets and certain periods of time (Metghalchi, Chang, & Marcucci, 2008; Schulmeister, 2009; Holmberg, Lönnbark, & Lundström, 2012; Lu & Shiu, 2016; J. S. Kim, D. H. Kim, & Seo, 2017; Nazario, Lima e Silva, Amorim OSbreiro, & Kimira, 2017; Vaiciulyte, Kalsyte, Sakalauskas, & Plikynas, 2017), so that it is possible to consistently beat the market in the mean-variance sense: improving market's profitability without worsening the risk.

Focusing on the technical analysis approach, many studies have been published which use it in order to define short and medium term investment strategies. The aim of this kind of analysis is to forecast the evolution of the price of financial assets using the information contained in their past prices. To this end, a wide range of indicators and oscillators are applied, together with chart pattern analysis. Literature studying the success of investment strategies making use of technical analysis is very extensive and continues to grow. Just to mention some recent examples, we can cite the papers by Ansari and Khan (2012) and Wu, Wang, and Chung (2016), which apply momentum strategies, Metghalchi et al. (2008) and Pla-María and García (2016), implementing moving average strategies, and Cervelló-Royo, Guijarro, and Michniuk (2015) and Arevalo, García, Guijarro, and Peris (2017), using chart pattern recognition strategies.

Other researchers have focused on volatility. For example, Hyup (2007) analyses the KSE KOSPI index volatility with different neural networks and Dixit and Roy (2013) apply back-propagation neural network to forecast the volatility of the Indian Stock Volatility Index. Other authors have used econometric volatility models from the GARCH family, as Dash and Dask (2016), who apply an evolutionary hybrid fuzzy computationally efficient EGARCH model.

Some short term investment strategies need to forecast the closing price of the next trading day. These strategies have recently generated a number of studies which try to predict next day's closing price by applying technical indicators together with other methodologies, such as deep learning (Gao, Li, Chai, & Tang, 2016). Many of these studies include the use of neural networks (Göçken, Özçalici, Boru, & Dosdogru, 2016; Su & Cheng, 2016; Wu & Duan, 2016; Moghaddam et al., 2016).

A similar approach consists on simply predicting stock price evolution during the next session. That is, the closing price of the next trading day is not to be forecast, but only the market trend. Then, a straightforward trading strategy to apply is the following: if an upward trend is predicted for the next trading session, a long position is opened at market opening; if a bear market is expected, a short position is opened. At the end of the day, positions are

closed. This approach has captured the attention of many academics. Among the pioneering studies some of the most outstanding are Leung, Daouk, & Chen (2000), Kim and Han (2000), Kim (2003), Bekiros (2010) and Kara, Boyacioglu, and Baykan (2011). These papers make use of different methodologies such as neural networks, genetic algorithms, support vector machines, discriminant analysis and logistic regression models to forecast stock market movement direction. More recent studies also apply such methodologies. Qiu and Song (2016) employ an optimized artificial neural network model, Inthachot, Boonjing, and Intakosum (2016) use artificial neural network and genetic algorithm hybrid intelligence, and Jabbarzadeh, Shavvalpour, Khanjarpanah, and Dourvash (2016) apply nonlinear probability models for predicting stock price trend.

The aim of this paper is likewise to forecast the market trend for the next trading day. More specifically, the purpose is to predict whether the price of the German DAX-30 stock index will increase or decrease at the closing of next day's trading session. As in most of the previous research papers analysed, a machine learning method is applied which is fed with information from technical analysis indicators and oscillators. In our case, a hybrid neuronal network is applied. One of the main differences of our proposal is, obviously, the methodology employed to forecast the market trend of next trading day. Another difference is that the proposed model is applied in a European stock index, whereas all the recent papers cited above analyse Asian and US indices such as S&P 500, Nikkei 225, SET50 (Thailand), KLCI Composite (Malaysia) etc. A third important issue is the dimensionality reduction of the indicators and oscillators. Due to the noise that the high number of inputs can introduce in the neural network, we propose to reduce the size of the inputs through a factor analysis on the indicators and oscillators. Finally, the hit ratio obtained in our study significantly improves the results obtained by most previous studies.

The remainder of the paper is organized as follows. Section one proposes a hybrid fuzzy neural network to predict the market trend for intraday trading. Section 2 describes the database and the experimental design. Section 3 shows the results obtained and compares them with those in previous studies. Finally, the concluding remarks are presented.

1. Prediction model

In recent years, a large number of studies apply artificial neural network (ANN) approaches for predicting stock market trend. Many studies combine ANNs with other approaches to increase predictions' accuracy. In this paper, we present a combination of artificial intelligence and fuzzy logic.

Hybrid fuzzy neural networks have two main advantages. First, the use of fuzzy rule-based systems and their ability to tackle uncertainty and non-linearity problems; and second, the use of a neural network in the learning process of the fuzzy rules of the system. The application of neural networks improves the learning process of the fuzzy rules from the data, so that no external expert knowledge is required during the process.

Hybrid models which combine artificial intelligence and fuzzy logic have been already successfully applied to solve different issues, as shows the literature survey by Mohd Adnan, Sarkheyli, Mohd Zain, and Haron (2015) and can beat other machine learning models such as backpropagation neural networks and other fuzzy models.

1.1. Fuzzy rule-based systems

Fuzzy rule-based systems are based on the fuzzy set theory (Zadeh, 1965), whose purpose is to use fuzzy IF-THEN rules to represent the knowledge of human experts. Later, other alternatives were developed to generate the fuzzy rule-based model without the participation of human experts. These methods generate the models automatically applying learning methods. There are a number of proposals to undertake this learning task such as gradient descendent learning methods (Ichihashi & Watanabe, 1990), space partition based methods (Wang & Mendel, 1992), heuristic procedures (Ishibuchi, Nozaki, & Tanaka, 1994) and neural-fuzzy techniques (Jang, 1993; Kim & Kasabov, 1999) among others.

In addition to selection of the learning method to be applied, the fuzzy reasoning structure has to be selected, as well. Depending on the types of fuzzy reasoning and fuzzy IF-THEN rules employed, most neuro-fuzzy inference systems can be classified into three types (Jang, 1993): Mamdani type, Takagi-Sugeno type or TSK and Tsukamoto type. In our research, we use the Mamdani model (Mamdani, 1974; Mamdani & Assilian, 1975), whose main advantages are its interpretability, as it uses linguistic variables, and its classification ability.

In the Mamdani model, fuzzy IF-THEN rules have the following form:

$$\text{If}\left(X_1 \text{ is } A_1 \text{ and} \dots \text{and } X_n \text{ is } A_n \right) \text{then } Y \text{ is } B, \tag{1}$$

where X_i and Y are, respectively, input and output linguistic variables and A_i and B are linguistic values.

The standard architecture of the Mamdani model consists of four components: fuzzification, knowledge base, inference engine and defuzzifier (Riza, Bergmeir, Herrera, & Benítez, 2015). The fuzzication component transforms the inputs into linguistic values. The knowledge base includes a database with the fuzzy set definitions and a rulebase containing the fuzzy IF-THEN rules. The inference engine undertakes the learning process using the fuzzy rules and the input data. Finally, the defuzzifier decodes the linguistic value into an output value.

1.2. Fuzzy rule-based systems and neural networks

Fuzzy neural networks (Buckley & Hayashi, 1994) combine artificial neural networks with fuzzy rule-based systems. A fuzzy rule-based system is laid upon the structure of a neural network and the learning algorithm of the latter is used to adapt the parameters in the fuzzy system. Among the different methods available to combine fuzzy systems and neural networks, in this research we will use the hybrid neural fuzzy inference system (HyFIS) introduced by Kim and Kasabov (1999).

HyFIS is a multilayer neural network-based fuzzy system with a total of five layers. The semantic meaning and functions of the nodes in the network are explained below, following Kim and Kasabov (1999).

Nodes in layer 1 are input nodes that represent input linguistic variables. Each node in the layer is connected to only those nodes of layer 2 which represent linguistic values of corresponding linguistic variable.

Nodes in layer 2 act as membership functions to represent the terms of the respective linguistic variables. Input values are fed to the layer 2, which calculates the membership degrees. This is implemented using Gaussian membership functions with two parameters: centre or mean (c) and width or variance (σ^2).

$$\mu A(x) = \text{Gaussian}(x; c, \sigma) = e^{-\frac{(x-c)^2}{\sigma^2}}. \tag{2}$$

Initially, a connection weight in this layer is the unity and the membership functions are spaced equally over the weight space. The output function of this node is the degree to which the input belongs to the given membership function:

$$y_i^{II} = e^{-\frac{(x-c)^2}{\sigma^2}}, \tag{3}$$

where σ^2 and c are the parameters. As the values of these parameters change, the bell-shaped functions vary, thus exhibiting various forms of membership functions on linguistic label.

In layer 3 each node represents a possible IF-part of a fuzzy rule. The weights of the links are set to unity. The nodes in this layer perform the AND operation. Therefore, all the nodes in this layer form a fuzzy rule base. The functions of the layer are as follows:

$$y_j^{III} = \min_{i \in I_j}\left(y_i^{II}\right), \tag{4}$$

where I_j is the set of indices of the nodes in layer 2 that are connected to node j in layer 3, and y_i^{II} is the output of node i in layer 2.

A node in layer 4 represents a possible THEN-part of a fuzzy rule and each node of this layer performs the fuzzy OR operation to integrate the field rules leading to the same output linguistic variables. The activation of the node represents the degree to which this membership function is supported by all fuzzy rules together. The initial connection weights of the links between layer 3 and 4 are randomly selected in the interval [−1, +1]. The functions of this layer are expressed as follows:

$$y_k^{IV} = \max_{j \in I_k}\left(y_j^{III} w_{kj}^2\right), \tag{5}$$

where I_k is the set of indices of the nodes in layer 3 that are connected to the node k in layer 4. Each of the rules is activated to a certain degree represented by the squared weight values.

Layer 5 represents the output variables of the system. These nodes and links attached to them act as defuzzifier. A node in this layer computes a crisp output signal using the Centre of Gravity method:

$$y_l^V = \frac{\sum_{k \in I_l} y_k^{IV} \sigma_{lk} c_{lk}}{\sum_{k \in I_k} y_k^{IV} \sigma_k}, \tag{6}$$

where I_t is the set of indices of the nodes in layer 4 which are connected to the node l in layer 5 and c_{lk} and σ_{lk} are respectively the centroid and width of the membership function of the output linguistic value represented by k in layer 4. The weights of the links from the nodes in layer 5 to the nodes in layer 4 are unity. Thus, the only learnable weights in the network are w_{kj}s between layer 3 and layer 4.

The learning procedure in HyFIS consists of two phases. The first phase is the structure learning (rule finding) phase using the knowledge acquisition module. The second phase is the parameter learning phase for tuning membership functions to achieve a desired level of performance. One advantage of this approach is the ease to modify the fuzzy rule base as new data become available (Wang & Mendel, 1992).

In the rule finding phase, a set of fuzzy rules from the desired input/output pairs is generated. Then these fuzzy rules are used to determine the structure of the neuro-fuzzy system in the HyFIS environment. So, in this phase, the first step consists of dividing the input and output space into fuzzy regions. Then fuzzy rules are generated from given data pairs. Finally, a degree is assigned to each rule.

In the parameter learning phase, after the fuzzy rules are established, the whole network structure is defined. Then, the network enters the second learning phase to optimally adjust the parameters of the membership functions. In this learning process, the gradient learning algorithm is employed to minimise error function E:

$$E = \frac{1}{2}\sum_{x}\sum_{l=1}^{q}\left(d_l - y_l^V\right)^2,$$ (7)

where q is the number of nodes in layer 5 and d_l and y_l^V are the target and actual outputs of node l in layer 5 for an input X.

Figure 1 shows the architecture of the HyFIS. For a more accurate description of HyFIS please refer to Kim and Kasabov (1999).
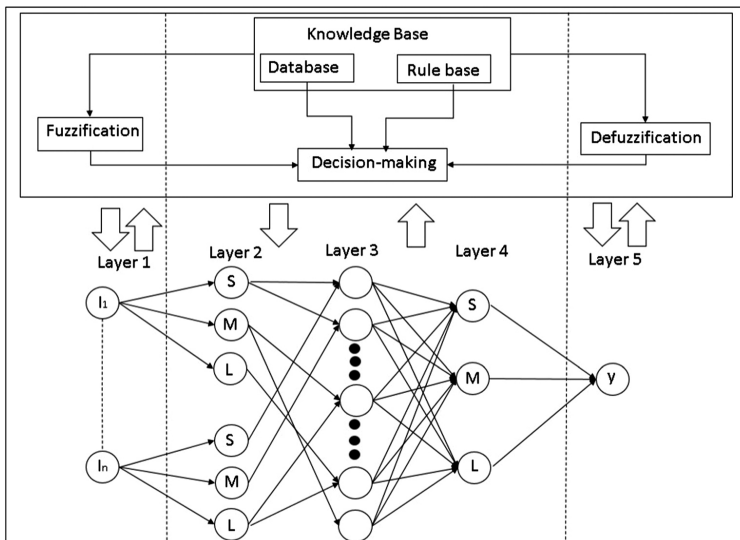

Figure 1. Architecture of the HyFIS, being S (small), M (medium) and L (large) linguistic variables

2. Database and experimental design

The HyFIS model described above has been applied to predict next trading day's trend in the German DAX-30 stock index. Our database includes a total of 4,345 trading days in the period from 8th December 1999 to 30th January 2017. For each trading day, information about opening, low, high and closing price is available. We divide the database in two groups: (1) the training sample, which lasts from December 1999 to November 2014 and contains 87% of the data, and (2) the testing sample, including the remaining trading days, which are used as out-of-sample data (Figure 2).

The period analysed includes bull as well as bear periods, which is an important feature in order to test the robustness of the prediction model. From January 2015 until April 2015 the market experienced an upward trend. Then, the trend turned and the index reached a local minimum in October 2015. Then, it sharply rose until December 2015 and again strongly fall during the next three months. From March 2016 to March 2017, the index rose again, experiencing a sideways trend from October 2016 to January 2017. Figure 2 clearly shows the high volatility in the German stock market during the period analysed.

The information in the database is used to calculate the technical analysis indicators and oscillators described in Table 1. As already mentioned in the introduction, it is very common to use technical analysis indicators to predict price evolution. In fact, all research papers cited include among the input variables some technical indicators. These previous studies have been considered to select the indicators in our research.

Figure 2. German DAX-30 performance during the period analysed
(8th December 1999 to 30th January 2017)

Table 1. Selected variables employed in the study

Indicator	Description	Equation
$A/OOSCILLATOR(n)_t$	Accumulation/ Distribution ($n = 1$)	$(H_t - C_{t-n})/(H_t - L_t)$
$ASY(n)_t$	Mean of rising days ($n = 2, 5, 10$)	$\left(\sum_{i=1}^{n} SY_{t-i+1}\right)/n$
$CCI(n)_t$	Commodity channel index ($n = 20$)	$(M_t - SM(n)_t)/(0.015 * D(n)_t)$
$MA(n)_t$	Moving average n periods ($n = 10, 20$)	$\left(\sum_{i=1}^{n} C_{t-i+1}\right)/n$
$MACD(n)_t$	Moving average convergence divergence ($n = 9$)	$MACD(n)_{t-1} + \dfrac{2}{n+1}\left(DIFF_t - MACD(n)_{t-1}\right)$ $DIFF_t = EMA(12)_t - EMA(26)_t$
$MOMENTUM(n)_t$	Momentum ($n = 4$)	$C_t - C_{t-n}$
$OBV(n)_t$	On balance volume ($n = 1$)	$\begin{cases} OBV_{t-n} + V_t & \text{if } C_t > C_{t-n} \\ OBV_{t-n} - V_t & \text{if } C_t < C_{t-n} \\ OBV_{t-n} & \text{if } C_t = C_{t-n} \end{cases}$
$PSY(n)_t$	Ratio of rising days ($n = 2, 5, 10$)	$A(n)_t/n$
$RSI(n)_t$	Relative strength index ($n = 14$)	$100 - 100/\left(1 + \dfrac{\sum_{i=0}^{n-1} Up_{t-1}}{n}/1 + \dfrac{\sum_{i=0}^{n-1} Dw_{i-1}}{n}\right)$
$LW\%R(n)_t$	William's percentage range ($n = 14$)	$100 \times (H_n - C_t)/(H_n - L_n)$

In Table 1, indicator's name, a brief description of the indicator and the formula to calculate it are presented. H_t, L_t and C_t stand for the highest price, the lowest price and the closing price at time t, respectively. H_n and L_n stand for the highest price and the lowest price in the last n periods, respectively. V_t represents the volume in period t.

The return of the index is calculated as $SY_t = ln(C_t/C_{t-1})$, while $ASY(n)_t$ computes the average return in the last n trading days. The CCI indicator is computed through the values of $M_t = (H_t + L_t + C_t)/3$, $SM(n)_t = \left(\sum_{i=1}^{n} M_{t-i+1}\right)/n$ and $D(n)_t = \left(\sum_{i=1}^{n} |M_{t-i+1} - SM_t|\right)/n$.

$A(n)_t$ is defined for period t as the number of rising days in the las n days. EMA stands for Exponential Moving Average. Finally, in the computation of the RSI indicator we use Up_t as the upward price change of the index at time t and Dw_t as the downward price change of the index at time t.

The indicators described in Table 1 can be used as inputs in the predictive models which apply artificial neural networks. Nevertheless, previous literature results show that an excessive number of inputs can introduce too much noise during the learning process. As a result, network's training could be affected by over optimization, especially if the input variables are correlated. In this case, the neural network could reach a high performance with the training data, but obtain bad forecasts in the testing period.

To cross-check this possibility, a model has been used which includes as inputs of the neural network all variables in Table 1.

In order to avoid potential over optimization issues in the neural network, factor analysis is applied to reduce the dimensionality of the indicators. Doing this, several experiments have been implemented in which the neural network has been fed with the factors resulting from the factor analysis, not with the original indicators in Table 1.

The factors obtained from the factor analysis are a linear combination of the original indicators. Varimax rotation has been applied to guarantee that the dimensions defined by the factors are different and independent of each other.

As shown in Figure 3, there are 3 factors which obtain an eigenvalue bigger than the unity. Therefore, it can be stated that all the information captured by the indicators in Table 1 can be efficiently summarized by these 3 factors.

Factors loadings in Table 2 unveil the correlation between each technical indicator and the 3 factors extracted. The first factor groups 8 of the 15 indicators. Among these 8 indicators there are those which theoretically forecast medium term market evolution (PSY and ASY for $n = 5, 10$), while those forecasting short term direction are included in factor 2 (PSY and ASY for $n = 2$). Other middle term indicators are assigned to factor 1 as well, as RSI, CCI, LW%R and MACD. Finally, factor 3 includes the Moving Average indicators ($n = 5, 10$) and the volume indicator OBV.

In the next section we will analyse whether the dimension reduction has a positive impact on price trend forecast results and a related trading strategy. Furthermore, we will study whether the use of 3 factors is the best option or if it is possible to obtain better results by using a different number of factors (i.e., not considering only those factors with eigenvalue greater than one).
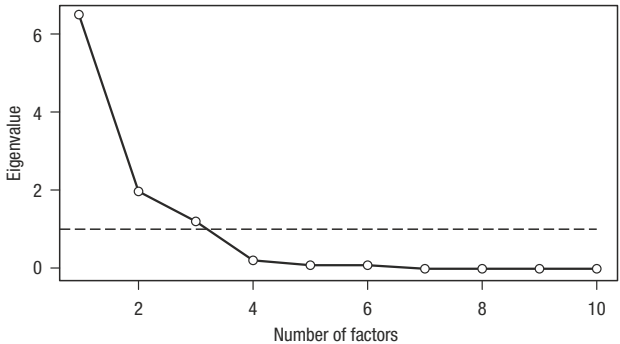

Figure 3. Eigenvalues of the factors obtained in the factor analysis

Table 2. Factor loadings

Indicator	Factor 1	Factor 2	Factor 3
RSI	**0.91745**	0.19189	0.01820
CCI	**0.90212**	0.24213	−0.02312
ASY(n = 10)	**−0.86524**	−0.17615	0.01236
LW%R	**−0.83574**	−0.38589	−0.00892
MACD	**0.77961**	−0.17888	0.02918
PSY(n = 10)	**0.77218**	0.18049	0.05286
ASY(n = 5)	**−0.66452**	−0.47684	0.02608
PSY(n = 5)	**0.61698**	0.45492	0.02560
MOMENTUM	0.10493	**0.81842**	−0.00351
ASY(n = 2)	−0.30677	**−0.80281**	0.02712
PSY(n = 2)	0.29763	**0.74748**	0.00961
A/OOSCILLATOR	0.04426	**0.73376**	0.03679
MA(n = 10)	0.01677	−0.00180	**0.98363**
MA(n = 5)	0.03065	0.00232	**0.98322**
OBV	0.00342	0.02379	**0.92396**

As in Qui and Song (2016), original data are normalized before being subjected to the neuro fuzzy system. The purpose is to avoid larger input values to overwhelm smaller ones.

Once the required data that feed the HyFIS model are obtained, the prediction of the direction for the next day of the DAX 30 index is performed. Using the training sample lasting from January 2000 to November 2014, the prediction model is estimated. Then this estimation model is applied on the remaining data. Equation 8 shows the forecast $\hat{y}_t$ for day t

$$\hat{y}_t = f\left(x_{t-1}\right), \tag{8}$$

where x_t is a vector representing the input variables used in the hybrid model.

It must be noted that the forecast indicates if the index is supposed to go up or down the next day. The percentage of correct predictions for a period $t = 1, \ldots, n$ is calculated by the hit ratio:

$$\text{Hit ratio} = \frac{1}{n}\sum_{t=1}^{n} P_t, \tag{9}$$

where P_t denotes the prediction for the tth trading day as defined in (10) and n denotes the number of test samples.

$$P_t = \begin{cases} 1, & \left(y_{t+1} - y_t\right)\left(\hat{y}_{t+1} - \hat{y}_t\right) > 0, \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where y_t is the actual closing price for the tth trading day and $\hat{y}_t$ is the predicted value for the same day.

3. Results and comparison with previous studies

This section shows the results obtained by the methodology described in the previous section when applied on the testing sample. First, Table 3 presents the hit ratio obtained by the different models.

Model A uses all technical indicators available in Table 1. Models B to E use as inputs the factors obtained in the factor analysis. The number of factors included in the prediction model varies (2, 3, 5 and 10 factors).

Table 3. Analysis of the hit ratio (HR) for the different prediction models

Model	Inputs for the HyFIS model	Hit ratio
Model A	All technical indicators	28%
Model B	2 Factors	74.03%
Model C	3 Factors	76.24%
Model D	5 Factors	73.48%
Model E	10 Factors	72.37%

Table 3 reveals how important it is to correctly select the input variables that feed the model in order to obtain an accurate prediction. It is especially striking that Model A, which employs all technical indicators as input variables, obtains the poorest results and is not able to reach a hit ratio of 50%. A naïve model to randomly predict next day's market trend should have a hit ratio of around 50%, considering that the probability of an upward and a downward trend is the same. Therefore, model A clearly reveals an important over optimization problem.

The model B including only 2 factors reaches a hit ratio of 74.03%. Including 3 factors (Model C) increases the hit ratio to 76.24%. This is the highest hit ratio obtained implementing factor models on the selected sample. In fact, when increasing the number of factors (Models D and E), the hit ratio decreases.

As stated above, directly including traditional technical indicators into the prediction model (Model A) does not generate good forecast results. The reason may be that the aim of technical indicators is not to predict one day ahead market trend. Each indicator is used in a different trading context. Usually a combination of two indicators is required to set a strategy. Therefore, applying them in a different context and with a different aim does not produce positive results.

In fact, in most of the research papers cited in Table 4, input variables only include traditional technical indicators. And in most cases the hit ratio is rather poor. Only the research by Kara et al. (2011) and Qui and Song (2016) employ not just traditional technical indicators but include other inputs like the PSY ratio. It must be stressed that those papers are precisely the only ones in the literature which obtain a hit ratio which is similar to ours.

Once the prediction models have been obtained, it is possible to implement a simple trading strategy with them. In fact, the final aim of the models is to take advantage of the forecasts to earn money in the stock exchange market, not just reaching a high hit ratio.

Table 4. Comparison of results with previous studies

Research	Methodology	Stock market	Hit ratio (%)
Leung et al. (2000)	Classification model	Japan	68
Kim and Han (2000)	GA feature discretion	Korea	61.7
Kim (2003)	Support Vector Machines	Korea	57.83
Bekiros (2010)	Neurofuzzy model	Malaysia	53.4
Kara et al. (2011)	BPNN	Turkey	75.74
Inthachot et al. (2016)	GA-ANN	Thailand	63.6
Jabbarzadeh et al. (2016)	Nonlinear probability models	USA	60.73
Qui and Song (2016)	GA-ANN	Japan	81.27
Our study	HyFIS	Germany	76.24

The strategy is as follows: if an upward trend is predicted for the next trading session, a long position is opened at market opening; if a bear market is expected, a short position is opened. At the end of the day, the position is closed, and the earnings or the loses are collected. This simple strategy has been used in previous studies such as Wu et al. (2016).

Figure 4 shows the cumulative wealth achieved by the strategy in the out-of-sample period, from December 2014 to January 2017, when implementing Model A (all technical indicators are included), Model B (two factors) and Model C (three factors). For clarity reasons, we omit plotting the results obtained by models D and E. In any case, they are in line with those by Model B and Model C. Nevertheless, it must be stressed that regardless the similar hit ratio of the factor models, the evolution of the cumulative wealth may be quite different during the period analysed. Figure 4 reveals that such is the case for models B and C.

Figure 4. Cumulative wealth of the investment strategy vs. DAX-30

Moreover, Figure 4 plots the evolution of the benchmark index DAX-30 during the same period. Figure 4 clearly shows that the return of the three proposed strategies beats the return of the German index, regardless the prediction model applied. This result is more obvious with models B and C.

Apart from this positive result regarding returns, Table 5 shows that the strategy beats the market in terms of risk, as well. The standard deviation of returns is 0.1384 for the DAX-30, whereas in the case of Model C (3 factors) it is 0.01097. Therefore, the Sharpe ratio is significantly better for the strategy (0.07023) than for the benchmark index (0.02646).

Table 5. Return and risk for the index and the strategies bases on our model

	Return DAX-30	Return A	Return B	Return C	Return D	Return E
Mean	0.00037	0.00028	0.00052	0.00077	0.00068	0.00030
St. Dev.	0.01384	0.01099	0.01098	0.01097	0.01098	0.01099
Sharpe	0.02646	0.02546	0.04741	0.07023	0.06174	0.02722

In the light of the results obtained, it can be concluded that applying HyFIS together with factor analysis to select the input factors to feed the artificial network is a satisfactory alternative to forecast next day's market trend. Furthermore, the predictions obtained can successfully be introduced into trading strategies that beat the market, both in terms of return and risk.

4. Discussion

It must be noted that the results presented in the previous section may be biased, as only one training data subsample and one testing subsample have been used. Nevertheless, the use of a reduced number of subsamples is usual in the literature. Recent previous studies divide the data in two parts (Kara et al., 2011; Jabbarzadeh et al., 2016; Qui & Song, 2016), three parts (Bekiros, 2010) or five parts (Inthachot et al., 2016).

In order to guarantee for the robustness of the results, we randomly train and apply the models 10,000 times. Each training sample contains 100 randomly selected observations out of the 4,345 trading days in the data base. Once the model is trained, it is applied to forecast the price direction of 15 randomly selected training days, which are used as out-of-sample data. This procedure is repeated 10,000 times.

As each model has been applied 10,000 times on different subsamples, Table 6 shows descriptive statistical information for each model.

Model B including only 2 factors reaches a mean hit ratio of 75.27%. This is the highest mean hit ratio obtained implementing factor models. Including 3 factors (Model C) slightly reduces the mean hit ratio to 74.85%. When increasing the number of factors to 5 (Model D), the hit ratio again decreases. Model E obtains a slightly better mean hit ratio of 75.07%. These outcomes are in line with the hit ratios in Table 3.

Table 6. Analysis of the hit ratio (HR) for the different prediction models

Model	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	Sd.
Model A	0.0000	0.2214	0.2399	0.2231	0.2565	0.3137	0.0660
Model B	0.6624	0.7399	0.7528	0.7527	0.7657	0.8303	0.0292
Model C	0.6716	0.7343	0.7491	0.7485	0.7620	0.8210	0.0393
Model D	0.6734	0.7343	0.7472	0.7477	0.7620	0.8210	0.0331
Model E	0.6716	0.7380	0.7509	0.7507	0.7657	0.8100	0.0210

It must be stressed that the small difference in the mean hit ratio obtained by models B to E is not an evidence than those models with higher hit ratio are better predictors. In fact, looking at the standard deviations in Table 6, we can conclude that the hit ratios of the models come from distributions with the same mean. Therefore, it can be stated that models B to E have the same prediction accuracy in statistical terms.

Nevertheless, having similar hit ratios does not mean that investing strategies applying the different forecast models should obtain the same return.

Conclusions

This paper applies a HyFIS model to forecast the daily evolution of the German Dax-35 stock index. The accurate one day ahead prediction is crucial for the correct performance of several trading strategies. For this reason, this issue has been addressed in many previous studies.

The results obtained in our study are very promising. In fact, we have reached a 76.24% mean hit ratio in the period analysed. This outcome is considerably better than those in most previous studies. Previous surveys apply different prediction methodologies on other stock indices, mainly Asian ones. Furthermore, the inputs that feed those models are limited to technical analysis indicators and oscillators. In our study, we have concluded that directly including traditional technical indicators and oscillators into the prediction model is not useful to perform the task of predicting the market trend one day ahead. This result is in line with that obtained in Qiu and Song (2016). We should note that most technical analysis indicators are not used on a stand-alone basis. Usually, they generate buy and sell signals when the graph line of two different indicators cross. And it must be underlined that these signals do not necessarily appear at the end of each trading session, rather during the session. Additionally, traditional indicators are not specifically designed to address the problem of forecasting next day trend. Finally, introducing many different technical indicators may generate too much noise in the prediction model together with over optimization .

This paper has highlighted the paramount importance of proper input variables selection. Our dimensionality reduction proposal is able to improve the result both for the hit ratio and the investment strategy. This means that the hit ratio of previous studies could probably be enhanced simply by changing the input variables that feed the models, as the lower hit ratio may not be a consequence of the methodologies applied, but of the inputs selected. In this line, input variables should not be limited to traditional technical indicators. Furthermore, dimension reduction techniques such as factor analyses can be helpful to avoid over optimization problems.

We have designed a simple investment strategy using the outcome of the prediction models. The result is very promising, as the strategy does not only beat the benchmark index in terms of returns, but also in risk.

In order to guarantee for the robustness of the results, we have randomly trained and applied the forecast models 10,000 times. This is a major difference compared with previous studies. As a result, we can conclude that there is no statistically significant difference in the mean hit ratio obtained by the different factor models.

Nevertheless, our study has got some limitations. The most important issue is the hit ratio obtained. Although it is better than the hit ratio in most previous studies, it is necessary to improve it. The variables employed in the model are selected from previous studies in order to facilitate the comparison of the results, but a deeper research is required to find variables with higher predictive power. By considering variables derived from price, the model can confront the weak form of the efficient market hypothesis. But this is not enough if we want to confront the semi-strong and the strong forms of the efficient market hypothesis. Furthermore, the investment strategy used in our paper is very simple, which is interesting for academic purposes, but not for professional investors. Finally, the prediction model is applied on just one stock index, so we do not know whether the forecast ability will remain in other markets. It may be interesting to investigate whether the hit ratio can be related to the market maturity, so testing the hypothesis that emerging markets can be less efficient than mature ones.

In line with the statements above, there are several issues that should be addressed in future research. First, new input variables could be used to improve the predictive ability of the model and its hit ratio. New variables will not arise from technical analysis, but it will be necessary to be innovative. Maybe, those capturing short term market sentiment could be a good start. Second, it is interesting to compare the different forecast methodologies proposed in the literature using the new input variables. Finally, in order to validate results robustness, it is necessary to apply the prediction models on different stock indices.

Disclosure statement

The authors declare that they have no competing financial, professional, or personal interests from other parties.

References

Ansari, V. A., & Khan, S. (2012). Momentum anomaly: evidence from India. *Managerial Finance*, *38*(2), 206-223. https://doi.org/10.1108/03074351211193730

Arevalo, R., García, J., Guijarro, F., & Peris, A. (2017). A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting. *Expert Systems with Applications*, *81*, 177-192. https://doi.org/10.1016/j.eswa.2017.03.028

Bekiros, S. D. (2010). Fuzzy adaptative decision-making for boundedly rational traders in speculative stock markets. *European Journal of Operational Research*, *202*, 285-293. https://doi.org/10.1016/j.ejor.2009.04.015

Buckley, J. J., & Hayashi, Y. (1994). Fuzzy neural networks: a survey. *Fuzzy Sets and Systems*, *66*(1), 1-13. https://doi.org/10.1016/0165-0114(94)90297-6

Cervelló-Royo, R., Guijarro, F., & Michniuk, K. (2015). Stock market trading rule based on pattern recognition and technical analysis: forecasting the DJIA index with intraday data. *Expert Systems with Applications*, *42*(14), 5963-5975. https://doi.org/10.1016/j.eswa.2015.03.017

Dash, R., & Dask, P. K. (2016). An evolutionary hybrid fuzzy computationally efficient EGARCH model for volatility prediction. *Applied Soft Computing*, *45*, 40-60. https://doi.org/10.1016/j.asoc.2016.04.014

Dixit, G., & Roy, D. (2013). Predicting India Volatility Index: an application of artificial neural network. *International Journal of Computer Applications*, *70*(4), 22-30. https://doi.org/10.5120/11950-7768

Fama, E. (1970). Efficient capital markets: a review of theory and empirical work. *Journal of Finance*, *25*(2), 383-417. https://doi.org/10.2307/2325486

Gao, T., Li, X., Chai, Y., & Tang, Y. (2016). *Deep learning with stock indicators and two-dimensional principal component analysis for closing price prediction system*. IEEE Xplore. https://doi.org/10.1109/ICSESS.2016.7883040

García, F., Guijarro, F., & Moya, I. (2011). The curvature of the tracking frontier: a new criterion for the partial index tracking problem. *Mathematical and Computer Modelling*, *54*, 1781-1784. https://doi.org/10.1016/j.mcm.2011.02.015

García, F., Guijarro, F., & Moya, I. (2013). A multiobjective model for passive portfolio management: an application on the S&P 100 index. *Journal of Business Economics and Management*, *14*, 758-775. https://doi.org/10.3846/16111699.2012.668859

Guijarro, F. (2017). A similarity measure for the cardinality constrained frontier in the mean-variance optimization model. *Journal of the Operational Research Society*, *69*(6), 928-945. https://doi.org/10.1057/s41274-017-0276-6

Göçken, M., Özçalici, M., Boru, A., & Dosdogru, A. (2016). Integrating metaheuristics and artificial neural networks for improved stock price prediction. *Expert Systems with Applications*, *44*, 320-331. https://doi.org/10.1016/j.eswa.2015.09.029

Holmberg, U., Lönnbark, C., & Lundström, C. (2012). Assessing the profitability of intraday opening range breakout strategies. *Finance Research Letters*, *10*(1), 27-33. https://doi.org/10.1016/j.frl.2012.09.001

Hyup, T. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications*, *33*, 916-922. https://doi.org/10.1016/j.eswa.2006.08.001

Ichihashi, H., & Watanabe, T. (1990). Learning control system by a simplified fuzzy reasoning model. *Journal of Japan Society for Fuzzy Theory and Systems*, *2*(3), 429-437. https://doi.org/10.3156/jfuzzy.2.3_429

Inthachot, V., Boonjing, V., & Intakosum, S. (2016). Artificial neural network and genetic algorithm hybrid intelligence for predicting Thai stock price index trend. *Computational Intelligence and Neuroscience*, *2016*, ID 3045254, 1-8. https://doi.org/10.1155/2016/3045254

Ishibuchi, H., Nozaki, K., & Tanaka, H. (1994). Empirical study on learning in fuzzy systems by rice taste analysis. *Fuzzy Sets and Systems*, *64*(2), 129-144. https://doi.org/10.1016/0165-0114(94)90329-8

Jabbarzadeh, A., Shavvalpour, S., Khanjarpanah, H., & Dourvash, D. (2016). A multiple-criteria approach for forecasting stock price direction: nonlinear probability models with application in S&P 500 Index. *International Journal of Applied Engineering Research*, *11*(6), 3870-3878.

Jang, J. S. (1993). ANFIS: adaptative-netowrk-based fuzzy inference systems. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665-685. https://doi.org/10.1109/21.256541

Kara, Y., Boyacioglu, M. A., & Baykan, Ö. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, *38*(5), 11-19. https://doi.org/10.1016/j.eswa.2010.10.027

Kim, J., & Kasabov, N. (1999). HyFIS: adaptative neuro-fuzzy inference systems and their application to nonlinear dynamical systems. *Neural Networks*, *12*(9), 1301-1319. https://doi.org/10.1016/S0893-6080(99)00067-2

Kim, K. J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, *55*(1), 307-319. https://doi.org/10.1016/S0925-2312(03)00372-2

Kim, K. J., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications*, *19*, 125-132. https://doi.org/10.1016/S0957-4174(00)00027-0

Kim, J. S., Kim, D. H., & Seo, S. W. (2017). Individual mean-variance relation and stock-level investor sentiment. *Journal of Business Economics and Management*, *18*(1), 20-34. https://doi.org/10.3846/16111699.2016.1252794

Leung, M. T., Daouk, H., & Chen, A. (2000). Forecasting stock indices: a comparison of classification and level estimation models. *International Journal of Forecasting*, *16*(2), 173-1990. https://doi.org/10.1016/S0169-2070(99)00048-5

Lim, K. P., Brooks, R. (2011). The evolution of stock market efficiency over time: A survey of the empirical literature. *Journal of Economic Surveys*, *25*(1), 69-108. https://doi.org/10.1111/j.1467-6419.2009.00611.x

Lu, T. H., & Shiu, Y. M. (2016). Can 1-day candlestick patterns be profitable on the 30 component stocks of the DJIA? *Applied Economics*, *48*(35), 3345-3354. https://doi.org/10.1080/00036846.2015.1137553

Mamdani, E. H. (1974). Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institution of Electrical Engineers*, *121*(12), 1585-1588. https://doi.org/10.1049/piee.1974.0328

Mamdani, E. H., & Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, *7*(1), 1-13. https://doi.org/10.1016/S0020-7373(75)80002-2

Metghalchi, M., Chang, Y. H., & Marcucci, J. (2008). Is the Swedish stock market efficient? Evidence from some simple trading rules. *International Review of Financial Analysis*, *17*(3), 475-490. https://doi.org/10.1016/j.irfa.2007.05.001

Moghaddam, A. H., Moghaddam, M. H., & Esfandyari, M. (2016). Stock market index prediction using artificial neural network. *Journal of Economics, Finance and Administrative Science*, *21*, 89-93. https://doi.org./10.1016/j.jefas.2016.07.002

Mohd Adnan, M. R. H., Sarkheyli, A., Mohd Zain, A., & Haron, H. (2015). Fuzzy logic for modelling machining process: a review. *Artificial Intelligence Review*, *43*, 345-379. https://doi.org/10.1007/s10462-012-9381-8

Nazario, R. T., Lima e Silva, J., Amorim OSbreiro, V., & Kimira, H. (2017). A literature review of technical analysis on stock markets. *The Quarterly Review of Economics and Finance*, *66*, 115-126. https://doi.org/10.1016/j.qref.2017.01.014

Pla-María, M., & García, F. (2016). Stock market investment implementing technical analysis strategies: moving average crossover. *Finance, Markets and Valuation*, *1*, 35-50.

Qiu, M., & Song, Y. (2016). Predicting the direction of stock market index movement using an optimized artificial neural network model. *PLoS ONE*, *11*(5). https://doi.org/10.1371/journal.pone.0155133

Riza, S. L., Bergmeir, C., Herrera, F., & Benítez, J. M. (2015). Fuzzy rule-based systems for classification and regression in R. *Journal of Statistical Software*, *65*(6), 1-30. https://doi.org/10.18637/jss.v065.i06

Schulmeister, S. (2009). Profitability of technical stock trading: has it moved from daily to intraday data? *Review of Financial Economics*, *18*(4), 190-201. https://doi.org/10.1016/j.rfe.2008.10.001

Su, C. H., & Cheng, C. H. (2016). A hybrid fuzzy time series model based on ANFIS and integrated nonlinear feature selection method for forecasting stock. *Neurocomputing*, *205*, 264-273. https://doi.org/10.1016/j.neucom.2016.03.068

Vaiciulyte, I., Kalsyte, Z., Sakalauskas, L., & Plikynas, D. (2017). Assessment of market reaction on the share performance on the basis of its visualization in 2D space. *Journal of Business Economics and Management*, *18*(1), 309-318. https://doi.org/10.3846/16111699.2017.1285348

Wang, L. X., & Mendel, J. M. (1992). Generating fuzzy rules by learning from examples. *EEE Transactions on Systems, Man, and Cybernetics*, *22*(6), 1414-1427. https://doi.org/10.1109/21.199466

Wu, B., & Duan, T. (2017). A performance comparison of neural networks in forecasting stock price trend. *International Journal of Computational Intelligence Systems*, *10*, 336-346. https://doi.org/10.2991/ijcis.2017.10.1.23

Wu, M. E., Wang, C. H., & Chung, W. H. (2016). Using trading mechanisms to investigate large futures data and their implications to market trends. *Soft Computing*. https://doi.org/10.1007/s00500-016-2162-6

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X