D. Bernardo, H. Hagras & E.P.K. Tsang, An interval Type-2 fuzzy logic based system for model generation and summarization of arbitrage opportunities in stock markets, 12th Annual UK Workshop on Computational Intelligence (UKCI), Edinburgh, UK, 5-7 September 2012, 1-7 (ISBN 978-1-4673-4391-6)

An Interval Type-2 Fuzzy Logic Based System for Model Generation and Summarization of Arbitrage Opportunities in Stock Markets

Dario Bernardo, Hani Hagras, Edward Tsang
The Computational Intelligence Centre
School of Computer Science and Electronic Engineering
University of Essex
Colchester, UK

***Abstract*—Today stock market exchange and finance are centers of attention all over the world. In finance, arbitrage is the practice of taking advantage of a price misalignment between two or more stock markets where profit can be earned by striking a combination of matching deals that capitalize upon the misalignment. If one strikes when misalignment has been observed, such deals are practically risk-free. However, when risk-free profit is around, everyone would compete to take advantage of it. Therefore, the question is whether arbitrage opportunities can be predicted; after all, misalignment does not happen instantaneously. Furthermore, financial operators do not like black boxes in forecasting. In this paper, we will present a type-2 Fuzzy Logic System (FLS) for the modeling and prediction of financial applications. The proposed system is capable of generating summarized models from pre-specified number of linguistic rules, which enables the user to understand the generated models for arbitrage opportunities prediction. The system is able to use this summarized model for the prediction of arbitrage opportunities in stock markets. We have performed several experiments based on the arbitrage data which is used in stock markets to spot ahead of time arbitrage opportunities. The proposed type-2 FLS has outperformed the Evolving Decision Rule (EDR) procedure (which is based on Genetic Programming (GP) and decision trees). Like GP, the type-2 FLS is capable of providing a white box model which could be easily understood and analyzed by the lay user.**

Keywords-component; Type-2 Fuzzy logic Systems; Financial Applications; arbitrage; prediction.

I. Introduction

Today stock market exchange and finance are centers of attention. However, the big number of the variables (and the associated complexity and randomness associated with these variables) affecting the stock market makes developing stock markets prediction models a very difficult task.

In economics and finance, arbitrage is the practice of taking advantage of a price misalignment between two or more assets and striking a combination of matching deals that capitalize upon the misalignment. When used by academics, an arbitrage is a transaction that involves no negative cash flow at any probabilistic or temporal state and a positive cash flow in at least one state; in simple terms, it is the possibility of a risk-free profit at zero cost [1]. Another definition of arbitrage is according to [2], where an arbitrage is the simultaneous purchase and sale of an asset in order to profit from a difference in the price. It is a trade that profits by exploiting price differences of identical or similar financial instruments, on different markets or in different forms. Arbitrage exists as a result of market inefficiencies; it provides a mechanism to ensure prices do not deviate substantially from fair value for long periods of time [2].

When price misalignment has occurred, many investors would be able to take advantage of it, especially when risk-free profit is feasible. Therefore, it is of great interest for companies and investors to identify arbitrage situations ahead of others. Being able to identify quickly arbitrage situations allows an investor to make an easy profit. However, developing accurate models for the prediction of arbitrage opportunities is a challenging task, this is because there are always high levels of uncertainties and risks associated with arbitrage opportunities, including minor risks (such as fluctuation of prices which could decrease the profit margins) and major risks (such as devaluation of a currency or derivative). In addition, arbitrage situation do not occur very often, and only last for seconds before the market adjusts itself. It is very important then to seize these scarce opportunities very quickly.

The problem with the time constraints is not only of being able to exploit the arbitrage situation on time, but being able to do it ahead of others, because otherwise the market will adjust before even spotting the situation [3]. For today's stock market with the availability of the internet and all the technology available, it is possible to sell and buy in milliseconds. For exploiting arbitrage situations we do not aim to analyze the status of the markets for evaluating if "at the moment" there are arbitrage opportunities. What we aim to do is to predict if in the next few minutes arbitrage opportunities will occur, in order to always be one step before others [3].

The other difficulties faced with the prediction of arbitrage opportunities are the scarce cases where the arbitrage chances occur [3]. Since these opportunities do not happen very often, the data will present relatively few samples representing the arbitrage situations. The scarcity of arbitrage chances makes chance discovery particularly difficult in machine learning, as only a very small percentage of the data is of significant interest. The challenge is actually to be able to identify those scarce cases. The prediction of the minority class in imbalanced data sets is a problem tackled in the Machine Learning field [4], [5], [6], [7]. There have been various approaches to detect this problem as many real world problems besides financial applications require the detection of rare cases as in the detection of oil spillage [8], fraud detection [9] and illnesses prediction [10], [11].

Input to a machine learning program determines the potential of finding useful patterns. In 2005, through close collaboration between economists and computer scientists an effective set of features were identified to enable arbitrage detection [3]. The dataset formation was not simply data translation and manipulation, but it is the work of months made by the research team on identifying the most relevant feature in the stock market and analyzing huge number of transactions. From millions of records, valuable information was retrieved for training variables and data were better pre-processed to help machine learning. New features were build substituting old ones by building complex formulas based on different stock market indicators. In [3] a method called EDDIE-ARB was used in order to identify arbitrage situations by analyzing option and futures prices in the London International Financial Futures Exchange (LIFFE) market. In [12], [13], [14] it was stated that theoretically arbitrage profit opportunities can be calculated from these features identified in [3].

The majority of the commercial financial tools employ the statistical regression techniques which capture only that information which can be refined into mathematical models to generate two outputs. Moreover, the regression techniques provide black box models which cannot be easily understood and analyzed by finance experts. Similarly advanced machine learning and artificial intelligence techniques like Neural Networks suffer from the same problem where they can give good prediction accuracies, however they provide black box models which are very difficult to understand and analyze by a finance expert. This causes the rare applications of these models where investors do not trust these black box models even if usually such models can give better performances than other methods. Hence, the ability of providing a clear and easy explanation has become today a very important requirement in financial applications.

Fuzzy Logic Systems (FLSs) provide white box models which could be easily analyzed and understood by the layman user. However FLSs suffer from the curse of dimensionality problem which causes the FLS based system to generate a big number of rules in order to give good model accuracy [15]. Most recently type-2 FLSs that are capable of handling high uncertainty levels have been employed for the generation of classification models [15], [16]. However, the existing type-2 fuzzy classification systems are not suited for the financial domain where such type-2 FLSs generate big rule bases and make the assumption that all the possible rules are represented in the existing models which is impossible for the huge financial data sets where the generated model will only cover a small subset of the search space. In this paper, we will present a type-2 FLS for the modeling and prediction of arbitrage opportunities in financial applications. The proposed system avoids the drawbacks of the existing type-2 fuzzy classification systems where the proposed system is able to carry prediction based on a relativity small pre-specified rule base size even if the incoming data vector does not match any rules in the FLS rule base. We have compared the proposed type-2 fuzzy logic based system with the Evolving Decision Rule (EDR) procedure [17] which is based on Genetic Programming and decision trees models and which represent some sort of a white box model. We will show through experiments on the LIFFE market how the proposed type-2 FLS has outperformed the EDR procedure while the type-2 FLS is capable of providing a white box model which could be easily understood and analyzed by financial experts. The proposed type-2 FLS aims to increase the understandability of the generated model by achieving the best performance with a limited and summarized number of rules in order to achieve simplicity and comprehensibility for the user . In Section II, we shall present a brief overview on type-2 FLSs and fuzzy classification systems. Section III will present an overview on the fuzzy classification systems. Section IV will present the proposed type-2 fuzzy based modeling and prediction system for financial applications. Section V will present the experiments on the arbitrage data and the achieved results. Finally Section VI will present the conclusions and future work.

II. Brief Overview on Type-2 Fuzzy Logic Systems and Fuzzy Classification Systems

A. Brief Overview on Type-2 Fuzzy Logic Systems

In the recent years type-2 FLSs have grown in popularity due to their ability to handle high levels of uncertainties. Type-2 FLSs employ type-2 fuzzy sets as shown in Fig. 1 where a type-2 fuzzy set is characterized by a fuzzy membership function, i.e. the membership value (or membership grade) for each element of this set is a fuzzy set in [0,1], unlike a type-1 fuzzy set where the membership grade is a crisp number in [0,1] [18]. The membership functions of type-2 fuzzy sets are three dimensional and include a footprint of uncertainty (shaded in grey in Fig. 1, it is the new third-dimension of type-2 fuzzy sets and the Footprint Of Uncertainty (FOU) that provide additional degrees of freedom that make it possible to directly model and handle uncertainties [18], [19]. The interval type-2 FLSs use interval type-2 fuzzy sets (such as the type-2 fuzzy set shown in Fig. 1 to represent the inputs and/or outputs of the FLS). In the interval type-2 fuzzy sets all the third dimension values are equal to one. The use of interval type-2 FLS helps to simplify the computation (as opposed to the general type-2 FLS).

The proposed system in the paper is a type-2 fuzzy classification system and hence it does not follow the structure of the type-2 FLSs reported in [18], [19] where the classification system process is summarized in the following section.

An interval type-2 fuzzy set denoted $\tilde{A}$ is written as follows:

$$\mu_{\tilde{A}}(x) = \int_{x\in X} \int_{u\in\left[\overline{\mu}_{\tilde{A}}(x),\ \underline{\mu}_{\tilde{A}}(x)\right]} 1/u \qquad (1)$$

$\overline{\mu}_{\tilde{A}}(x)$, $\underline{\mu}_{\tilde{A}}(x)$ represent the upper and lower membership functions respectively of the interval type-2 fuzzy set $\tilde{A}$. The upper membership function is associated with the upper bound of the footprint of uncertainty $FOU(\tilde{A})$ of a type-2 membership function. The lower membership function is associated with the lower bound of $FOU(\tilde{A})$ [18].

In our system, the generation process of the employed interval type-2 fuzzy sets starts by generating type-1 fuzzy sets which equally partition the input universe of discourse into a given number of partitions. We then blur the type-1 fuzzy sets to the left and the right equally by a given uncertainty factor as shown in Fig. 1a to generate the type-2 fuzzy sets. In the application shown in this paper, we have employed 4 fuzzy sets as shown in Fig. 1b to represent each input variable

B. *A Brief Overview on Fuzzy Logic Classification Systems*

In fuzzy logic classification systems, for a given c-class pattern classification problem with *n* attributes (or features), a given rule in the FLS rule base could be written as follows:

Rule R^j*: If* x_1 *is* A_1^j *and … and* x_n *is* A_n^j *then Class* C_j *with* CFj, $j = 1,2,\dots,N$ (2)

Where $x_1,\dots,x_n$ represent the n-dimensional pattern vector, A_i^j is the fuzzy set representing the linguistic label for the antecedent pattern *i*, C_j is a consequent class (which could be one of the possible c classes), *N* is the number of fuzzy if-then rules in the FLS rule base. CF_j is a certainty grade of rule j (i.e., rule weight). In case each input pattern is represented by *K* fuzzy sets and given that we have *n* input patterns, the possible number of rules that will cover the whole search space is K^n. In the arbitrage application presented in this paper, we have 7 inputs where each input is represented by 4 fuzzy sets; hence the needed number of rules to cover the whole search space for this given application is 4^7 = 16384 rules. In our given application (which applies to the vast majority of financial applications), we do not have enough data to generate this huge number of rules. Hence, there will be various cases where the incoming input vector will not fire any rule in the FLS rule base.

Several type-1 fuzzy classification systems have been reported in the literature such as [20], [21], [22], [23], [24], [25], [26], [27], [28] and [29]. However, in the vast majority of these papers, the data was quite easy to partition, and if an input pattern does not match any of the decision areas previously labeled, the input is discharged. In financial applications this cannot be done where if a new pattern that has never been seen before is proposed, a decision needs to be made anyway, and unfortunately discharging it a priory cannot be the solution. A technique to resolve this problem was proposed in [3], [30] and [31], this technique keeps in a rule repository all the rules for the minority class in unbalanced data sets. All the inputs that do not match any rule in the repository are considered belonging to the majority class. This technique can work in unbalanced data set but might not work in all cases.

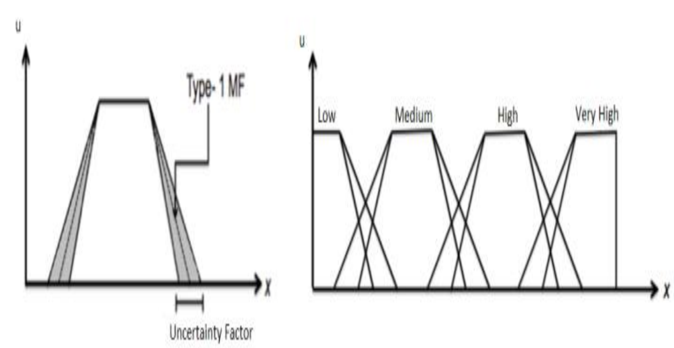

Figure 1. a) The process followed to generate a type-2 fuzzy set from a type-1 fuzzy set. b) The employed interval type-2 fuzzy sets in the application reported in this paper.

III. The Proposed Type-2 Fuzzy Modeling and Prediction System for ARBITRAGE PREDICTION

The proposed system has two phases, a modeling phase and a prediction phase. In the modeling phase the rule base of the type-2 fuzzy classification system is constructed from the existing training dataset. In the prediction phase, the generated rule base is used to predict the incoming input vectors.

A. *The modeling phase*

The modeling phase operates according to the following steps (as shown in Fig. 2):

Step 1- Raw Rule Extraction: For a fixed input-output pair ($x^{(t)}$, $C^{(t)}$) in the dataset, $t=1,\dots T$ (*T* is the total number of data training instances available for the modeling phase) compute the upper and lower membership values $\overline{\mu}_{A_s^q}$, $\underline{\mu}_{A_s^q}$ for each antecedent fuzzy set $q=1,\dots K$ (*K* is the total number of fuzzy sets representing the input pattern s where $s=1\dots n$.). Generate all rules combining the matched fuzzy sets A_s^q (i.e. either $\overline{\mu}_{A_s^q}>0$ or $\underline{\mu}_{A_s^q}>0$) for all $s=1\dots n$. Thus the rules generated by ($x^{(t)}$, $C^{(t)}$) will have different antecedents and the same consequent class $C^{(t)}$. Thus each of the extracted raw rules by ($x^{(t)}$, $C^{(t)}$) could be written as follows:

R^j *: If* x_1 *is* $\tilde{A}_1^{qjt}$ *and … and* x_n *is* $\tilde{A}_n^{qjt}$ *then Class* C_t, $t = 1,2,\dots,T$ (3)

For each generated rule, we calculate the firing strength F^t. This firing strength measures the strength of the points $x^{(t)}$ belonging to the fuzzy region covered by the rule. F^t is defined in terms of the lower and upper bounds of the firing strength $\underline{f^{(t)}}, \overline{f^{(t)}}$ of this rule which are calculated as follows:

$$\overline{f^{jt}}\left(x^{(t)}\right) = \overline{\mu_{A_1^{qjt}}}(x_1) * \cdots * \overline{\mu_{A_n^{qjt}}}(x_n) \quad (4)$$

$$\underline{f^{jt}}\left(x^{(t)}\right) = \underline{\mu_{A_1^{qjt}}}(x_1) * \cdots * \underline{\mu_{A_n^{qjt}}}(x_n) \quad (5)$$

The * denotes the minimum or product t-norm. Step 1 is repeated for all the t data points from 1 to *T* to obtain generated rules in the form of Equation (3). If there are two or more rules generated which have the same antecedents and consequent classes, we will aggregate these rules in one rule having the same antecedents and the same consequent class with the associated $\overline{f^{jt}}$ and $\underline{f^{jt}}$ which result in the maximum average $(\overline{f^{jt}} + \underline{f^{jt}})/2$ amongst these rules.

The financial data is usually highly imbalanced (as in the case of arbitrage situations where we have few arbitrage examples available in the data sets). Hence, we will present a new approach called "***scaled dominance***" which tries to handle imbalanced data by trying to increase the confidence and support for the minority class. In order to compute the scaled dominance for a given rule having a consequent Class C_j, we divide the firing strength of this rule by the summation of the firing strengths of all the rules which had C_j as the consequent class. This allows handling the imbalance of data towards a given class. We scale the firing strength by scaling the upper and lower bounds of the firing strengths as follows:

$$\overline{fs^{jt}} = \frac{\overline{f^{jt}}}{\sum_{j\in Classj}\overline{f^{j}}} \tag{6}$$

$$\underline{fs^{jt}} = \frac{\underline{fs^{jt}}}{\sum_{j\in Classj}\underline{fs^{jt}}} \tag{7}$$

Step 2- Scaled Support and Scaled Confidence Calculation: Many of the generated rules will share the same antecedents but different consequents. To resolve this conflict, we will calculate the scaled confidence and scaled support which are calculated by grouping the rules that have the same antecedents and conflicting classes. For given m rules having the same antecedents and conflicting classes. The scaled confidence ($\tilde{A}_q \Rightarrow C_q$) (defined by its upper bound $\overline{c}$ and lower bound $\underline{c}$, it is scaled as it involves the scaled firing strengths mentioned in the step above) that class C_q is the consequent class for the antecedents $\tilde{A}_q$ (where there are m conflicting rules with the same antecedents and conflicting consequents) could be written as follows:

$$\overline{c}\left(\tilde{A}_q \Rightarrow C_q\right) = \frac{\sum_{x_s\in\, Class\, C_q}\overline{fs^{jt}}(x_s)}{\sum_{j=1}^{m}\overline{fs^{jt}}(x_s)} \tag{8}$$

$$\underline{c}\left(\tilde{A}_q \Rightarrow C_q\right) = \frac{\sum_{x_s\in\, Class\, c_q}\underline{fs^{jt}}(x_s)}{\sum_{j=1}^{m}\underline{fs^{jt}}(x_s)} \tag{9}$$

The scaled confidence can be viewed as measuring the validity of $A_q \Rightarrow C_q$. The confidence can be viewed as a numerical approximation of the conditional probability [22]. The scaled support (defined by its upper bound $\overline{s}$ and lower bound $\underline{s}$, it is scaled as it involves the scaled firing strengths mentioned in the step above of $A_q \Rightarrow C_q$ is written as follows:

$$\overline{s}\left(\tilde{A}_q \Rightarrow C_q\right) = \frac{\sum_{x_s\in\, Class\, C_q}\overline{fs^{jt}}(x_s)}{m} \tag{10}$$

$$\underline{s}\left(\tilde{A}_q \Rightarrow C_q\right) = \frac{\sum_{x_s\in\, Class\, c_q}\underline{fs^{jt}}(x_s)}{m} \tag{11}$$

The support can be viewed as measuring the coverage of training patterns by $A_q \Rightarrow C_q$. In this paper we introduce the concept of scaled dominance, (defined by its upper bound $\overline{d}$ and lower bound $\underline{d}$) which is calculated by multiplying the scaled support and scaled confidence of the rule as follows:

$$\overline{d}\left(\tilde{A}_q \Rightarrow C_q\right) = \overline{c}\left(\tilde{A}_q \Rightarrow C_q\right)\cdot \overline{s}\left(\tilde{A}_q \Rightarrow C_q\right) \tag{12}$$

$$\underline{d}\left(\tilde{A}_q \Rightarrow C_q\right) = \underline{c}\left(\tilde{A}_q \Rightarrow C_q\right)\cdot \underline{s}\left(\tilde{A}_q \Rightarrow C_q\right) \tag{13}$$

For rules that share the same antecedents and have different consequent classes, we will replace these rules by one rule having the same antecedents and the consequent class which will be corresponding to the rule that gives the highest scaled dominance value. In [15], the rule generation system generates only the rule with the highest firing strength, however in our method, we generate all rules that are generated by the given input patterns, and this allows covering a bigger area in the decision space.

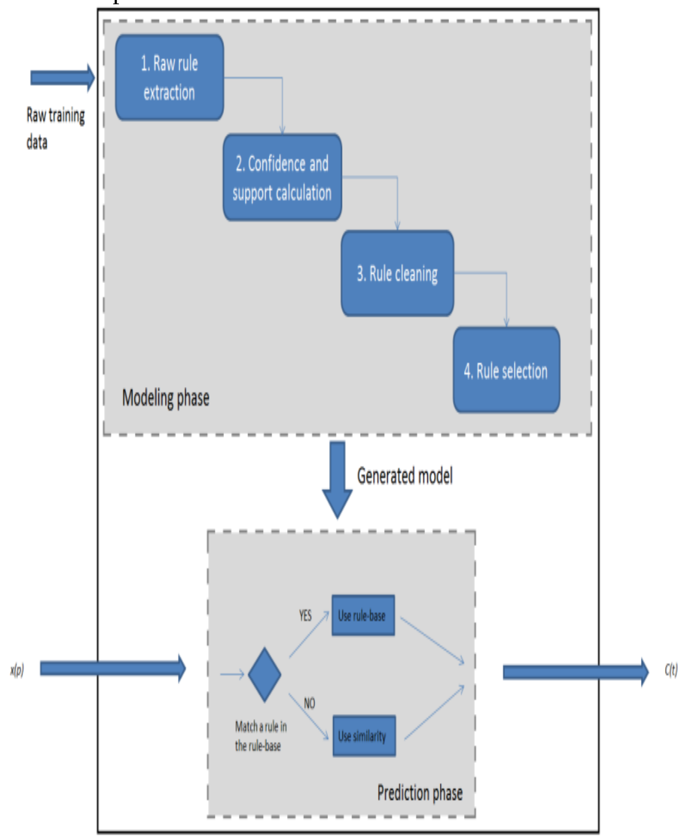

Figure 2. An overview of the proposed modeling and prediction system

Step 4- Rule Selection: As fuzzy based classification methods generate a large number of rules, this could cause major problems for financial applications where the users need to understand the system. Hence, in our method, we will reduce the rule base to a relatively small pre-specified size of rules that generates a summarized model which could be easily read, understood and analyzed by the user. In this step, we select only the top Y rules per class (Y is pre-specified by the given financial application) which has the rules with the highest scaled dominance values. This selection is useful because rules with low dominance may not actually be relevant and possibly introduce errors. This helps to keep the classification system more balanced between the majority and minority classes. By the end of this step, the modeling phase is finished where we have $X = nC \cdot Y$ rules (with nC the number of classes) ready to classify and predict incoming patterns as discussed below in the prediction phase.

B. Prediction Phase

When an input pattern is introduced to the generated model, two cases will happen: the first case is when the input $x^{(p)}$ matches any of the X rules in the generated model, in this case we will follow the process explained by *case 1*. If $x^{(p)}$ does not match any of the existing X rules, we will follow the process explained by *case 2*.

Case 1: The input matches one of the existing rules

In case the incoming input $x^{(p)}$ matches any of the existing X rules, we will calculate the firing strength of the matched rules according to Equations (4) and (5), this will result in $\overline{f^j}(x^{(p)})$, $\underline{f^j}(x^{(p)})$. In this case, the predicted class will be determined by calculating a vote for each class as follows:

$$\overline{Z}Class_h(x^{(p)}) = \frac{\sum_{j\in h}\overline{f^j}(x^{(p)})*\overline{d}(A_q\to C_q)}{\max j\in h(\overline{f^j}(x^{(p)})*\overline{d}(A_q\to C_q))} \quad (14)$$

$$\underline{Z}Class_h(x^{(p)}) = \frac{\sum_{j\in h}\underline{f^j}(x^{(p)})*\underline{d}(A_q\to C_q)}{\max j\in h(\underline{f^j}(x^{(p)})*\underline{d}(A_q\to C_q))} \quad (15)$$

The total vote strength is then calculated as:

$$ZClass_h = \frac{\overline{Z}Class_h(x^{(p)}) + \underline{Z}Class_h(x^{(p)})}{2} \quad (16)$$

The class with the highest $ZClass_h$ will be the class predicted for the incoming input vector $x^{(p)}$.

Case 2: The input does match any of the existing rules

In case the incoming input vector $x^{(p)}$ does not match any of the existing X rules, we need to find the closest rule in the rule base that matches $x^{(p)}$. In order to do this, we need to calculate the similarity (or distance) between each of the fuzzy rule generated by $x^{(p)}$ and each of the X rules stored in the rule base. The rules generated by $x^{(p)}$ are found by taking each element in $x^{(p)}$ and taking all matching fuzzy sets with either $\overline{\mu_{A_1^{qj}}}(x_i)$ or $\underline{\mu_{A_1^{qj}}}(x_i)$ greater than 0. At this point there will be k rules generated from the input $x^{(p)}$. Let the linguistic labels that fit $x^{(p)}$ be written as $v_{inputr} = (v_{input1r}, v_{input2r}, \ldots, v_{inputnr})$ where r is the index of the r-th rule generated from the input. Let the linguistic labels corresponding to a given rule in the rule base be $v_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$. Each of these linguistic labels (*Low, Medium, etc*) could be decoded into an integer. Hence the similarity between the rule generated by $x^{(p)}$ and a given rule in the rule base could be calculated by finding the distance between the two vectors as follows:

$$Similarity_{input\ r\leftrightarrow j} = \left(\left(1-\left|\frac{vinput1r-vj1}{V1}\right|\right)* \left(1-\left|\frac{vinput2r-vj2}{V2}\right|\right)*\ldots*\left(1-\left|\frac{vinputnr-vjn}{Vn}\right|\right)\right) \quad (17)$$

In the equation V_s represents the number of linguistic labels representing each variable s. Each rule in the rule-base will have at this point a similarity associated with the *r-th* rule generated form the input. In this case, the predicted class will be determined by firstly selecting the rules with the highest similarity with the *r-th* generated rule. There might be more than one rule with the same similarity. Considering the rules that will have the same similarity with the *r-th* rule, the winning class for the *r-th* generated rule is calculated as a vote for each class as follows:

$$\overline{Z}Class_{h\,r}(x^{(p)}) = \sum_{j\in h}\overline{d}(A_q\to C_q) \quad (18)$$

$$\underline{Z}Class_{h\,r}(x^{(p)}) = \sum_{j\in h}\underline{d}(A_q\to C_q) \quad (19)$$

The total vote strength is then calculated as:

$$ZClass_{h\,r} = \frac{\overline{Z}Class_{h\,r}(x^{(p)}) + \underline{Z}Class_{h\,r}(x^{(p)})}{2} \quad (20)$$

The class with the highest $ZClass_{h\,r}$ will be the class associated with the *r-th* rule generated from the input. From all the k rules generated from $x^{(p)}$, the final output class is calculated by applying Equations (14), (15) and (16).

IV. Experiments and Results

We have performed experiments based on the data which has been used for spotting arbitrage opportunities in the London International Financial Futures Exchange (LIFFE) market [31]. We have compared the proposed approach with one of the most powerful white box modeling and prediction systems for spotting arbitrage opportunities which is EDR procedure [31]. The EDR method evolves a set of decision rules by using Genetic Programming (GP) and it receives feedback from a key element that is called *repository*. The repository is a structure whose objective is to store a set of rules. The resulting rules are used to create a range of classifications that allows the user to choose the best trade-off between misclassifications and false alarms cost.

The type-2 FLS aims to fulfill two objectives: The first one is to get good results on both RECALL (fraction of relevant instances that are correctly retrieved or True Positive Rate) and false positive rate, the second objective is to use small number of rules to model and predict the arbitrage opportunities, thus presenting a white box model which could be easily understood and analyzed by the lay user. The type-2 FLS allows the expert to easily modify or add rules that can reflect the changes of legislation. This will help to present a complete framework which could be useful in different risky or risk averse scenarios. The perfect ideal classifier is able to have a RECALL (True Positive Rate) of 1 and a False Positive Rate (FPR) of 0. In general, this means having the highest RECALL possible and the lowest false positive rate possible. Moving along the Receiver Operating Characteristic (ROC) curve (which plots the true positive rate, vs. false positive rate) means increasing the FPR at the expenses of the RECALL or vice versa.

In the following experiments, we have employed a type-2 FLS with different configurations in order to move along the ROC curve. We used a type-2 FLC with different degree of uncertainty as well as a different number of rules.

Figure 3 shows the ROC curve obtained by the type-2 FLS plotted against the ROC curve obtained by the EDR procedure [31]. From Figure 3, it is obvious the type-2 FLS gives a much better ROC curve than the EDR procedure while the type-2 FLS presents the user with a small number of rules .which summarizes the model and explains the system behavior to the lay user in an understandable and comprehensible way.

The red circles in Figure 3 show the results obtained by employing the *scaled dominance* and only the most relevant 30, 20 and 10 rules from the whole rule-base. Higher points on the left and on the right sides are obtained using the full rule-base. The results shows that the system is able to summarize the rule base and get very good results. The results obtained using 10 rules gives an improvement of about 16% on Recall for the correspondent point on the EDR curve for the same FPR value. Thus the type-2 FLS improves the accuracy of prediction by 16 % when compared with the EDR procedure having the same FPR rate. With 20 rules the Recall improvement gets up to 22% and with 30 rules the Recall improvement gets up to 25% when compared with the EDR procedure at the same FPR rate.

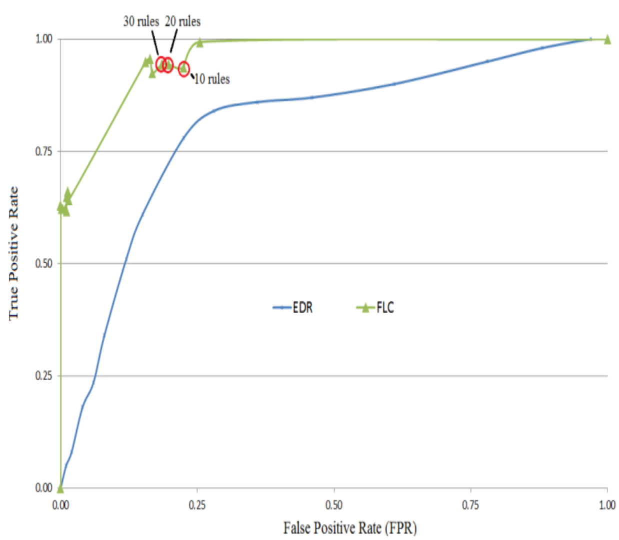

Figure 3. ROC curve of the type-2 FLS plotted against the ROC curve of the EDR procedure

V. CONCLUSIONS

In this paper we have presented a fuzzy type-2 logic system that is capable of handling the encountered uncertainties and generating summarized models of a relatively small number of pre-specified linguistic rules which enables the user to understand the generated model for arbitrage opportunities prediction. The system is able to use this summarized model for the prediction of arbitrage opportunities in stock markets. We have performed several experiments based on the arbitrage data which is used in stock markets to spot ahead of time arbitrage opportunities. We have performed experiments based on the arbitrage data which is used in the London International Financial Futures Exchange (LIFFE) market to spot ahead of time arbitrage and investment opportunities. We have compared the proposed approach with one of the most powerful white box modeling and prediction systems for spotting arbitrage opportunities which is EDR procedure [31]. It was shown that the type-2 FLS gives a much better performance than the EDR method in terms of accuracy of prediction and recall as was shown by the ROC curve while the type-2 FLS presents the user with a small number of rule which summarize the model and explain the system behavior to the lay user in an understandable and comprehensible way.

For our future work, we will use genetic algorithms in order to best tune the employed type-2 fuzzy sets. This will help to build a more flexible tool able to move along the ROC curve with more detail.

REFERENCES

[1] http://en.wikipedia.org/wiki/Arbitrage .

[2] http://www.investopedia.com/terms/a/arbitrage.asp .

[3] E.P.K. Tsang, S. Markose, H. Er, "Chance discovery in stock index option and future arbitrage", *New Mathematics and Natural Computation, World Scientific*, Vo.1, No.3, pp. 435-447, 2005.

[4] N. Japkowicz, "The class imbalance problem: Significance and strategies", *Proceedings of the 2000 International Conference on Artificial Intelligence (IC-AI'2000)*, vol. 1, pp. 111–117, 2000.

[5] G. M. Weiss, "Mining with rarity: A unifying framework, Special issue on learning from imbalanced datasets", vol. 6, pp. 7–19, 2004.

[6] G. E. A. P. A. Batista, R. C. Prati, M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data", *SIGKDD Explor. Newsl. 6*, no. 1, pp. 20–29, 2004.

[7] K. McCarthy, B. Zabar, G. Weiss, "Does cost-sensitive learning beat sampling for classifying rare classes?", *UBDM '05: Proceedings of the 1st international workshop on Utility-based data mining*, New York, NY, USA, ACM Press, pp. 69–77, 2005.

[8] M. Kubat, R. C. Holte, S. Matwin, "Machine learning for the detection of oil spills in satellite radar images", Machine Learning, vol. 30, pp. 195-215, 1998.

[9] T. Fawcett, F. J. Provost, "Adaptive fraud detection", *Data Mining and Knowledge Discovery 1*, no. 3, pp. 291–316, 1997.

[10] M. Greiner, D. Pfeiffer, R. D. Smith, "Principles and practical application of receiver-operating characteristic analysis for diagnostic tests", Prevent Veterinary Med 45, pp. 23–41, 2000.

[11] G. Vanagas, "Receiver operating characteristic curves and comparison of cardiac surgery risk stratification systems", *Interact Cardio-Vasc Thorac Surg 3*, no. 2, pp. 319–322, 2004.

[12] D. Modest, M. Sunderesan, "The relationship between spot and futures prices in stock index futures markets: some preliminary evidence", *Journal of Futures Markets*, Vol. 3, pp. 15-41, 1983.

[13] B. Cornell, K. French, "Taxes and the Pricing of Stock Index Futures", *Journal of Finance*, Vol. 38, pp. 675-694, 1988.

[14] A.L. Tucker, "Financial Futures, Options and Swaps", *West Publishing Company*, St. Paul, MN, 1991.

[15] J. Sanz, A. Fernandez, H. Bustince, F. Herrera, "Improving the performance of fuzzy rule-based classification systems with interval-valued fuzzy sets and genetic amplitude tuning". *Information Sciences*, 180, pp. 3674-3685, 2010.

[16] J. Sanz, A. Fernandez, H. Bustince, F. Herrera, "A genetic tuning to improve the performance of fuzzy Rule-Based Classification Systems with Interval-Valued fuzzy sets: Degree of ignorance and lateral position". *International Journal of Approximate Reasoning*, vol 52, pp. 751-766, 2011.

[17] A. L. Garcia-Almanza, E. P. K. Tsang, "Evolving decision rules to predict investment opportunities," *Int. J. Autom. Comput*., vol. 5, no. 1, pp. 22–31, Jan. 2008.

[18] H. Hagras, "A hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots". *IEEE Transactions on Fuzzy Systems*, vol 12, 4, pp. 524-539, 2004.

[19] J. Mendel, "Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions". *Prentice-Hall*, 2001.

[20] H. Ishibuchi, "Comparison of Heuristic criteria for fuzzy rule selection in classification problems". *Fuzzy Optimization and Decision making*, vol 3, 2, pp. 119-139, 2004

[21] H. Ishibuchi, "Effect of rule weights in fuzzy rule-based classification system". *IEEE Transactions on Fuzzy Systems*, vol 1, pp. 59-64 , 2000.

[22] H. Ishibuchi, “Rule Weight specification in fuzzy rule-based classification systems”. *IEEE Transactions on Fuzzy Systems*. Vol 13, 4, pp. 428-435, 2005.

[23] H. Ishibuchi, “Three-objective genetic-based machine learning for linguistic rule extraction”. *Information Sciences*. Vol 136, 1-4, pp. 109-133, 2001.

[24] H. Ishibuchi, T. Yamamoto, “Fuzzy rule selection by multy-objective genetic local search algorithms and rule evaluation measures in data mining”. *Information Sciences*. Vol 141, 1, pp. 59-88, 2004.

[25] H. Ishibuchi, T. Yamamoto, T. Nakashima, “An approach to Fuzzy Default Reasoning for function approximation”. *Soft Computing*. Vol 10, 9, pp. 850-864, 2006.

[26] A. Shigeo, M. Lan, “A method for fuzzy rules extraction directly from numerical data and its application in pattern classification”. *IEEE transactions on fuzzy systems*, vol 3, 1, pp. 18-28, 1995.

[27] S. Ahmad, M. Jahormi “Construction accurate fuzzy classification systems: A new approach using weighted fuzzy rules”. Computer Graphics, Imaging and Visualisation, pp. 408-413, 2007.

[28] L. Wang, “The WM Method Completed: A flexible fuzzy system approach to data mining”. IEEE Transactions on fuzzy systems, vol 11, 6, pp. 768-782, 2003.

[29] E. Mansoori, M. ZolGhadri, D. Katebi, “Using distribution of data to enhance performance of fuzzy classification system”. Iranian Journal of Fuzzy Systems, vol. 4, 1, 2006.

[30] A. L. Garcia-Almanza, “New classification methods for gathering patterns in the context of genetic programming”, PhD Thesis, Department of Computing and Electronic Systems, University of Essex, 2008.

[31] A. L. Garcia-Almanza, E. P. K. Tsang, “Forecasting stock prices using genetic programming and chance discovery”, 12th International Conference On Computing In Economics And Finance, 2006.