UNIVERSITY OF OSTRAVA
FACULTY OF SCIENCE
DEPARTMENT OF MATHEMATICS

LINGUISTIC APPROACH TO TIME SERIES FORECASTING

Ph.D. THESIS

AUTHOR: Mgr. Lenka Štěpničková
SUPERVISOR: prof. Ing. Vilém Novák, DrSc.

2018

OSTRAVSKÁ UNIVERZITA
PŘÍRODOVĚDECKÁ FAKULTA
KATEDRA MATEMATIKY

JAZYKOVÝ PŘÍSTUP K PREDIKCI ČASOVÝCH ŘAD

DOKTORSKÁ DISERTAČNÍ PRÁCE

AUTOR: Mgr. Lenka Štěpničková
VEDOUCÍ PRÁCE: prof. Ing. Vilém Novák, DrSc.

2018

Já, níže podepsaná studentka, tímto čestně prohlašuji, že text mnou odevzdané závěrečné práce v písemné podobě i na CD nosiči je totožný s textem závěrečné práce vloženým v databázi DIPL2.

Prohlašuji, že předložená práce je mým původním autorským dílem, které jsem vypracovala samostatně. Veškerou literaturu a další zdroje, z nichž jsem při zpracování čerpala, v práci řádně cituji a jsou uvedeny v seznamu použité literatury.

Ostrava ....

(podpis)

Beru na vědomí, že tato doktorská disertační práce je majetkem Ostravské univerzity (autorský zákon Č. 121/2000 Sb., §60 odst. 1), bez jejího souhlasu nesmí být nic z obsahu práce publikováno.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Ostravské univerzity.

Ostrava ....

(podpis)

Acknowledgements

I want to express my gratitude to my supervisor Prof.Vilém Novák for his support, valuable comments and permanent encouragement which made it possible to finish this thesis. Warm thanks goes to my husband for his infinite support and patience. I would also like to thank my colleagues from the Institute for Research and Applications of Fuzzy Modeling for making a friendly and creative atmosphere.

Ostrava, November 2018 Lenka Štěpničková

Summary

The time series forecasting plays an important role in supporting individual and organizational decision making which have wide practical use in economics, industry, meteorology and other areas. Besides many classical approaches to time series forecasting, several computational intelligence techniques have been proposed, e.g., Artificial Neural Network, evolutionary computation, Support Vector Machine or fuzzy techniques. Due to the wide variety of distinct methods for time series forecasting there always exists a danger of choosing a method that is inappropriate for a given time series. To overcome such a problem, distinct ensemble techniques, that combine several individual forecasts, were proposed.

The goal of this thesis is to propose novel approaches to time series forecasting. We introduce a new hybrid combination using linguistic fuzzy rules and the other computational intelligence methods. This hybrid model is easier to be interpret by decision-makers when modeling trended series. Further, we propose so-called Fuzzy Rule-Based Ensemble that is an ensemble technique combining distinct forecasting methods. Moreover, we focus on redundant rules and propose an algorithm for their automatic detection and removal.

The structure of this thesis is as follows. In Chapter 1, we introduce the concepts necessary for the understanding of the subsequent chapters. In Chapter 2, we present three approaches to multi-step seasonal time series forecasting: the Automatic Design of Artificial Neural Networks, the Support Vector Machine and the linguistic fuzzy approach. We propose hybrid combinations of these methods, such that the fuzzy approach to the trend-cycle forecasts is complemented by the earlier two approaches that forecast seasonal components. The experimental justification of their potential is proposed. In Chapter 3, we formally investigate how to determine fuzzy/linguistic IF-THEN rules that are redundant in linguistic descriptions. We present a formal definition of redundancy and show that seemingly redundant rules can be indispensable. An algorithm for the automatic detection and removal of redundant rules is also described. In Chapter 4, we introduce so-called Fuzzy Rule-Based Ensemble. This method is constructed as a linear combination of a small

number of forecasting methods where the weights are determined by fuzzy rule bases with time series features used as the antecedent variables. For the identification of fuzzy rule bases we use the linguistic association mining. A huge experimental justification is also provided.

Keywords: Time series, Fuzzy rules, Perception-based logical deduction, Ensemble techniques, Fuzzy rule-based ensemble, Linguistic associations, Fuzzy GUHA, Redundancy.

Anotace

Předpovídání časových řad hraje důležitou roli při individuálním a organizačním rozhodování, které má široké praktické využití v ekonomice, průmyslu, meteorologii a dalších oblastech. Kromě mnoha klasických přístupů k prognozování časových řad, bylo navrženo několik technik výpočetní inteligence, např. umělá neuronová síť, evoluční výpočet, metoda strojového učení nebo fuzzy techniky. Vzhledem k široké škále odlišných metod pro prognózu časových řad, vždy existuje nebezpečí volby metody, která je pro danou časovou řadu nevhodná. Proto byly navrženy různé kombinační metody, které kombinují několik individuálních předpovědí.

Cílem této práce je navrhnout nové přístupy k předpovídání časových řad. Představujeme novou hybridní kombinaci jazykových fuzzy pravidel a metod výpočetní inteligence. Tento hybridní model lze snadněji interpretovat při modelování časových řad s trendem. Dále představujeme kombinační techniku založenou na fuzzy pravidlech, která kombinuje různé předpovědní metody. Také se zaměřujeme na redundantní pravidla a nabízíme algoritmus pro jejich automatickou detekci a jejich následné odstranění.

Struktura této práce je následující. V kapitole 1 uvádíme pojmy nezbytné pro pochopení následujících kapitol. V kapitole 2 uvádíme tři přístupy k prognozování sezónních časových řad: umělé neuronové sítě, metoda strojového učení a jazykový fuzzy přístup. Nabízíme kombinaci těchto metod tím způsobem, že fuzzy přístup k předpovědi trendo-cyklu je doplněn dalšími dvěma metodami, které předpovídají sezónní složky. Abychom ukázali potenciál těchto metod, uvádíme i experimentální výpočty. V kapitole 3 formálně zkoumáme, jak určit fuzzy/jazyková IF-THEN pravidla, která jsou v jazykových popisech redundantní. Představujeme formální definici redundance a ukazujeme, že zdánlivě redundantní pravidla mohou být nepostradatelná. Také je popsán algoritmus pro automatickou detekci a odstranění redundantních pravidel. V kapitole 4 uvádíme kombinační techniku založenou na fuzzy pravidlech. Tato metoda je konstruována jako lineární kombinace malého počtu předpovědních metod, kde jsou váhy určeny bází fuzzy pravidel, jejichž antecedenty jsou tvořeny rysy časových řad. Pro identifikaci bází fuzzy pravidel

používáme jazykové dobývání dat. Rovněž nabízíme velkou experimentální studii.

Klíčová slova: Časová řada, Fuzzy pravidla, Fuzzy logická dedukce, Kombinační techniky, FRBE, Jazykové asociace, Fuzzy GUHA, Redundance.

Author's contribution

My contribution to the thesis can be summarized as follows:

- Chapter 2: I was significantly contributing to the whole idea of this collaborative research, to the creation of the fundamental ideas about linguistic approach to time series modeling as well as combining distinct computational intelligence methods in the presented way. This means to focus on the use of the robust and interpretable form of the linguistic approach to model the long-term dependencies mirrored in the trend-cycle and to use the power of support vector machine and/or adaptive neural networks to model the seasonal parts of the modeled time series. Substantial part of the experiments, the one devoted to the linguistic approach, was performed by myself as well.

- Chapter 3: I have participated on the formulation of the main research tasks and took part in the joint work on formalization of all the definitions. All the theorems and hypothesis, that are provided in this chapter, were formulated or proven with my personal participation.

- Chapter 4: I have initiated the first research goals in the Fuzzy Rule-Based Ensemble approach, formulated the tasks and goals, participated in structuring the ensemble into given "blocks". The idea of using linguistic associations mining for the given task in the graded form was also created with my contribution, my original experiments appeared in the earlier publications are not mirrored in the thesis as they were replaced by later experiments made by my co-authors.

List of Author's Publications

Publications in International Journals

- Štěpnička, M., Burda, M. and Štěpničková, L.(2016), Fuzzy Rule Base Ensemble Generated from Data by Linguistic Associations Mining. *Fuzzy Sets and Systems*, 285, 140–161.

- Dvořák, A., Štěpnička, M. and Štěpničková, L.(2015), On redundancies in systems of fuzzy/linguistic IF-THEN rules under perception-based logical deduction inference. *Fuzzy Sets and Systems*, 277, 22–43.

- Štěpnička, M., Cortez, P., Peralta Donate, J. and Štěpničková, L.(2013), Forecasting seasonal time series with computational intelligence: on recent methods and the potential of their combinations. *Expert Systems with Applications*, 40, 1981–1992.

- Štěpnička, M., Dvořák, A., Pavliska, V. and Vavříčková, L.(2011), A linguistic approach to time series modeling with the help of F-transform. *Fuzzy Sets and Systems*, 180, 164–184.

- Novák, V., Štěpnička, M., Dvořák, A., Perfilieva, I., Pavliska, V. and Vavříčková, L.(2010), Analysis of seasonal time series using fuzzy approach. *International Journal of General Systems*, 39, 305–328.

Publications in Conference Proceedings

- Burda, M., Štěpnička, M. and Štěpničková, L.(2015), Fuzzy Rule-Based Ensemble for Time Series Prediction: Progresses with Associations Mining. In: *Proc. Strengthening Links between Data Analysis and Soft Computing, Heidelberg*, 261–271.

- Štěpnička, M., Štěpničková, L. and Burda, M.(2014), Fuzzy Rule-Based Ensemble for Time Series Prediction: The Application of Linguistic Associations Mining. In: *Proc. IEEE International Conference on Fuzzy Systems, Beijing, China*, 505–512.

- Novák, V., Pavliska, V., Štěpnička, M. and Štěpničková, L.(2014), Time series trend extraction and its linguistic evaluation using F-transform and fuzzy natural logic. In *Recent Developments and New Directions in Soft Computing (Studies in Fuzziness and Soft Computing 317)*, 429–442.

- Štěpničková, L., Štěpnička, M. and Sikora, D.(2013), Fuzzy rule-based ensemble with use linguistic associations mining for time series prediction. In: *Proc. of the 8th Conference of EUSFLAT*, 408–415.

- Štěpničková, L., Štěpnička, M. and Dvořák, A.(2013), New results on redundancies of fuzzy/linguistic IF-THEN rules. In: *Proc. of the 8th Conference of EUSFLAT*, 400–407.

- Sikora, D., Štěpnička, M. and Vavříčková, L.(2013), On the potential of fuzzy rule-based ensemble forecasting. In: *Proc. of the International Joint Conference CISIS'12-SOCO'12 Special Sessions (Advances in Intelligent Systems and Computing)*. Springer-Verlag, Berlin, Heidelberg, 487–496.

- Sikora, D., Štěpnička, M. and Vavříčková, L.(2013), Fuzzy rule-based ensemble forecasting:introductory study. In: *Proc. of the Synergies of Soft Computing and Statistics for Intelligent Data Analysis (Advances in Intelligent Systems and Computing)*. Springer-Verlag, Berlin, Heidelberg, 379–387.

- Dvořák, A., Štěpnička, M. and Vavříčková, L.(2011), Redundancies in systems of fuzzy/linguistic IF-THEN rules. In: *Proc. of the 7th Conference of EUSFLAT and LFA-2011*, 1022–1029.

- Štěpnička, M., Peralta Donate, J., Cortez, P., Vavříčková, L. and Gutierrez, G.(2011), Forecasting seasonal time series with computational intelligence: contribution of a combination of distinct methods. In: *Proc. of the 7th Conference of EUSFLAT and LFA-2011*, 464–471.

- Sikora, D., Štěpnička, M. and Vavříčková, L.(2011), Combining techniques in time series forecasting. In: *Proc. of the Mendel2011-17th International Conference on Soft Computing*, 419–426.

- Štěpnička, M., Dvořák, A., Pavliska, V. and Vavříčková, L.(2010), Linguistic approach to time series analysis and forecasts. In: *Proc. of the 2010 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2010)*, 2149–2157.
- Štěpnička, M., Pavliska, V., Novák, V., Perfilieva, I., Vavříčková, L. and Tomanová, I.(2009), Time series analysis and prediction based on fuzzy rules and the fuzzy transform. In: *Proceedings of the Joint 2009 IFSA World Congress and 2009 EUSFLAT Conference*, 483–488.

Table of Contents

List of Figures

List of Tables

Chapter 1

Theoretical background

In this chapter we introduce some concepts necessary for the understanding of next chapters.

1.1 Introduction to time series analysis and forecasting

Forecasting the future is an important tool to support individual and organizational decision making. Time series forecasting predicts the behavior of a given phenomenon based solely on the past patterns of the same event. In particular, an interesting time series forecasting variant addresses seasonal data (e.g. monthly sales). Under such analysis, multi-step ahead prediction, i.e. forecasting several periods in advance, is highly relevant (e.g. for setting early production plans) in distinct domains, such as Agriculture, Finance, Sales and Production [82].

Before we introduce the used forecasting methods, we briefly recall the problem. A time series is usually given as a finite sequence $y_1, y_2, \ldots, y_T$ of real numbers. Formally, a time series can be defined as follows [93].

Definition 1 A time series is a function

$$y : \mathbb{T} \times \Omega \to \mathbb{R}, \tag{1.1.1}$$

where $\mathbb{T} = \{1, \ldots, T\} \subset \mathbb{N}$ is a finite set of integers interpreted as *time moments* and $\langle \Omega, \mathbb{A}, P \rangle$ is a probabilistic space, where Ω is a set of elementary events, $\mathbb{A}$ is a sigma algebra on Ω and P is a probabilistic measure defined on $\mathbb{A}$ such that $P(\Omega) = 1$.

If we fix some $\omega \in \Omega$ then we obtain a *realization* of the time series (1.1.1), which is a real discrete function y_t for $t = 1, \ldots, T$. Of course, in reality, we always have only one realization of y at our disposal.

Time series is defined in (1.1.1) on the domain $\mathbb{T} = \{1, \ldots, T\} \subset \mathbb{N}$. Let us now consider a set $\mathbb{T}^F = \{T+1, \ldots, T+h\}$. Then, forecasting of the time series y means to find its *future values* $\{y_t \mid t \in \mathbb{T}^F\}$ on the basis of the *known values* $\{y_t \mid t \in \mathbb{T}\}$. The number h is called *forecasting horizon*.

Now, we suppose that we are given the time series $\{y_t \mid t \in \mathbb{T} \cup \mathbb{T}^F\}$. The elements of the set $\{y_t \mid t \in \mathbb{T}\}$ are called *in-samples* and the elements of the set $\{y_t \mid t \in \mathbb{T}^F\}$ are called *out-samples*. Furthermore, we divide the in-samples $\{y_t \mid t \in \mathbb{T}\}$ into two subsets:

- learning set $\{y_t \mid t \in \mathbb{T}^L\}$ where $\mathbb{T}^L = \{1, \ldots, T^L\}$ for some $T^L < T$,
- validation set $\{y_t \mid t \in \mathbb{T}^V\}$ where $\mathbb{T}^V = \{T^L + 1, \ldots, T\}$.

On the basis of the learning set we find the best model of the time series, where we measure the quality of the model on the validation set. Finally, we compute the estimation $\{\hat{y}_t \mid t \in \mathbb{T}^F\}$ of the future values of the time series using the best model above. We have to stress that only in-samples are used to build such a model. The out-samples $\{y_t \mid t \in \mathbb{T}^F\}$ become the *testing set* using which we can measure how good is our forecast.

1.1.1 Classical forecasting methods

There are many distinct methods for time series analysis and forecasting. In this subsection, we briefly mention only some of them, namely those, that are used in the subsequent chapters. For more details about these methods, we refer to rich literature [14, 15, 54, 63]

Exponential smoothing methods that were proposed in the late 1950s [16, 57] belong to the most widely used forecasting methods. These techniques are very robust and work well even when the data is noisy, irregular or relatively short. The three most commonly used models are simple, Holt and Winter's. The models differ from one another in the way they treat trending and seasonality.

The most fundamental exponential smoothing methods is called simple exponential smoothing. This method is appropriate for time series with no trend and no seasonal patterns. The simple exponential smoothing model is given by the formula:

$$\hat{y}_t = \alpha y_t + (1 - \alpha)\hat{y}_{t-1},$$

where $\hat{y}_t$ is the smoothed value (forecast), $t \geq 3$ and $\alpha \in (0, 1)$ is a smoothing parameter.

The forecast is weighted average of the current value and the previous forecast value with the weights decreasing exponentially depending on the value of parameter α.

Holt exponential smoothing is an extension of exponential smoothing designed for time series with a trend. It is not appropriate for seasonal data. Forecasts extrapolate statistical estimates of the underlying level and trend at the end of the data. There are two smoothing parameters α and β. Holt exponential smoothing

model is given as follows:

$$\hat{y}_t = A_t + B_t,\ t = 1, 2, \ldots, T$$

where

$$A_t = \alpha y_t + (1 - \alpha)\hat{y}_{t-1}$$
$$B_t = \beta(A_t - A_{t-1}) + (1 - \beta)B_{t-1}$$

where A_t and B_t are (exponentially smoothed) estimates of the level and linear trend of the series at time t, respectively. The parameter α determines the level of the time series and the parameter β relates to its trend.

The forecasted values are determined as follows

$$y_{T+i} = A_t + kB_t,\ i = 1, \ldots, h.$$

The Winter's exponential smoothing is an extension of Holt exponential smoothing taking into account seasonality. The seasonality is captured in the form of seasonal indexes. There are two versions of the Winter's exponential smoothing: the multiplicative one and the additive one.

The Winter's multiplicative method is appropriate when a time series with a linear trend has a multiplicative seasonal component. The model of this method is given by the following equations:

$$A_t = \alpha\frac{y_t}{\hat{y}_{t-s}} + (1 - \alpha)(A_{t-1} + B_{t-1}),$$
$$B_t = \beta(A_t - A_{t-1}) + (1 - \beta)B_{t-1},$$
$$\hat{y}_t = \gamma\frac{y_t}{A_t} + (1 - \gamma)\hat{y}_{t-s}$$

where s is the number of period in one cycle of seasons, e.g., number of months or quarters in a year. Parameters α, β, γ take values from the interval $(0, 1)$.

The forecasted values are determined as follows

$$y_{T+i} = (A_t + kB_t)\hat{y}_{t-s+i},\ i = 1, \ldots, h.$$

The Winter's additive method, which is appropriate for time series with a linear trend with an additive seasonal component, is analogous.

Another classical approach stems from the Box-Jenkins methodology [14, 54] and it consists in autoregressive and moving average models. For instance, the ARMA(p,q) model is a typical representative of this methodology, which assumes that every value y_t of a given time series can be computed as follows:

$$y_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i y_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}, \tag{1.1.2}$$

where $\varphi_1, \ldots, \varphi_p$ are parameters of the autoregressive model; $\theta_1, \ldots, \theta_q$ are parameters of the moving average model; c is a constant; ε_t is a white noise term; and $\varepsilon_{t-1}, \ldots, \varepsilon_t - q$ are error terms.

The ARMA model (1.1.2), like most of the other Box-Jenkins methods, works under the stationarity assumption, i.e., assuming that the moments of y_t, such as mean and variance, do not change over time. This implicitly means that the time series should not contain any observable trend. To apply the standard Box-Jenkins approach to a trend-containing time series and, thus, take advantage of its powerful properties, one must first de-trend a given time series or apply an *autoregressive integrated moving average* ARIMA(p,d,q) model. The integrated part that has been added to the ARMA process can model generally polynomial trends. The parameter d determines the trend polynomial order; for example, $d = 1$ indicates a constant trend (with non-zero average), whereas $d = 2$ indicates a linear trend, $d = 3$ indicates a quadratic trend, and so on.

Let us mention that the ARIMA model has several variants; for example there

is a seasonal version (called SARIMA) and a fractional version (FARIMA), which allows parameter d to be a certain non-integer value.

A random walk [63] is another time series model where the current observation is equal to the previous observation with a random step up or down. First of all, a series of the first-order differences, i.e., the series of the changes between consecutive observations in the original series

$$\Delta y_t = y_t - y_{t-1}$$

is constructed. When such a series of the first-order differences performs as a white noise, the model for the original series can be written as

$$y_t - y_{t-1} = \varepsilon_t,$$

where ε_t denotes white noise. And thus, the *random walk* model is given as follows

$$y_t = y_{t-1} + \varepsilon_t.$$

Random walk model is widely used for non-stationary data, particularly financial and economic data. Random walk typically has the long periods of apparent trends up or down and the sudden and unpredictable changes in direction.

The forecasts from a random walk model are equal to the last observation, as future movements are unpredictable, and are equally likely to be up or down.

There are many other forecasting methods, ranging from very simple to very sophisticated. Dealing with all of them is far beyond the scope of this work. Thus, we mentioned only these methods that are used in experimental parts in the subsequent chapters.

1.1.2 Advanced forecasting methods

Besides many classical or say statistical approaches to time series forecasting, many attempts to apply advanced methods and models from distinct branches to this task have been provided. Here, we will only briefly mention some remarks related to the computational intelligence methods that provide a promising alternative that can enrich the existing powerful choice among statistical methods.

Computational intelligence denotes a branch of the Artificial Intelligence field that relies on heuristic algorithms inspired in biological and natural intelligence. These computational intelligence algorithms include elements of learning and adaptation (e.g., neural networks, fuzzy rules, evolutionary computation) that facilitate intelligent behavior in complex real-world problems [37].

Although mainly statistical time series forecasting methods (e.g., Holt-Winter's exponential smoothing or ARIMA methodology) are widely used in practice [82], several computational intelligence techniques have been proposed for forecasting as well [98]. For instance, some examples of computational intelligence applied to forecasting time series are: Artificial Neural Networks [30], evolutionary computation [27], Support Vector Machines [84], immune systems [97], fuzzy techniques [8], or their combinations [68, 99].

Focusing mainly on fuzzy approaches to time series analysis and forecast, it should be stressed that a significant number of works with this aim have been published. For instance, a study presenting Takagi-Sugeno rules [116] in view of the Box-Jenkins methodology has already been published, see [8]. However, the Takagi-Sugeno rules use functional consequents without any linguistic meanings, their antecedents are usually determined by a cluster analysis, and they do not employ any kind of logical implication. Thus, they can be considered a special kind

of regression model rather than a linguistic approach.

Let us also recall [118], where the authors directly fuzzify the ARIMA method by employing fuzzy numbers $\tilde{\varphi}_i, \tilde{\theta}_i$ instead of real numbers φ_i, θ_i as the model parameters.

Analogously, various neuro-fuzzy approaches, which lie on the border between neural networks, Takagi-Sugeno models and evolving fuzzy systems, are often successfully used [107, 76]. However, quite often, Gaussian or other types of fuzzy sets are tuned to have the center, say, at node 5.6989 and the width parameter equal to 2.8893 (see [76]), which is obtained using some optimization technique. The interpretability of such fuzzy sets is undoubtedly far from the interpretability of systems using models based on fragments of natural language.

Therefore, it may be reasonably stated that published approaches, although very effective and powerful, are so far closer to standard regression methods than to an interpretable linguistic approach.

A specific category is formed by the so-called *"fuzzy time series"* proposed in [115] and followed by many authors (see, e.g., [60]). This concept, briefly speaking, constructs fuzzy rules with fuzzy sets in antecedents as well as consequents (i.e., these models are linguistically motivated and constructed, in contrast to Takagi-Sugeno rules). However, neither these works employ any concepts from linguistics as a self-content scientific discipline, and they use mathematically incorrect notations. Furthermore, they employ the Mamdani-Assilian style of modeling fuzzy rules, i.e., this approach does not deal with genuine fuzzy IF-THEN rules of a conditional nature (called gradual rules [34] or implicative rules [120]) but with conjunctive rules.

It has to be stressed that works dealing with the concept of fuzzy time series

do not distinguish between the learning set plus the validation set and the testing set*). In other words, they measure forecasting precision on the same data that was used for the model identification. Consequently, they solve an approximation/interpolation problem rather than a time series prediction problem. Therefore, they are inappropriate for any comparison because they have never been demonstrated to perform the forecasting task.

It should be mentioned that while computational intelligence methods were successfully employed in different real-world tasks and several papers on their use in time series forecasting were published, they became more standard in data mining applications rather than in time series, where statistical methods still dominate the market [82]. Such preference for established statistical methods is due to several factors, such as conservatism of some forecasting community members [104], but mainly due to a heritage of inferior performance of the first attempts to apply computational intelligence to time series forecasting. Moreover, recent computational intelligence approaches to forecasting often ignore very important issues such as hyperparameter selection (e.g., optimal choice of Artificial Neural Networks topology), although it has been proved that an appropriate feature and model selection for a computational intelligence model is crucial in order to provide constantly better performance [29]. Similarly, some typical arguments in favor of fuzzy models, such as interpretability and linguistic nature of fuzzy models may seem to be a sort of an unsupported claim or even an empty cliché [12].

*)To the best of our knowledge, there is only one publication [77] dealing with the concept of fuzzy time series that clearly and correctly treats the in-samples and out-samples.

1.1.3 Accuracy measures

The global performance of a forecasting model is evaluated by an error measure. The standard measures of forecast accuracy are the Mean Absolute Error

$$\text{MAE} = \frac{1}{h}\sum_{t=T+1}^{T+h} |y_t - \hat{y}_t|, \tag{1.1.3}$$

the Mean Squared Error

$$\text{MSE} = \frac{1}{h}\sum_{t=T+1}^{T+h} (y_t - \hat{y}_t)^2, \tag{1.1.4}$$

and the Root Mean Squared Error

$$\text{RMSE} = \sqrt{\frac{1}{h}\sum_{t=T+1}^{T+h} (y_t - \hat{y}_t)^2}. \tag{1.1.5}$$

Historically, Mean Absolute Error (MAE) or (Root) Mean Squared Error ((R)MSE) are very popular error measures. However, Mean Square Error is too sensitive to outliers [7] and furthermore, both (R)MSE and MAE are scale-dependent measures and hence, it can be hardly used for a comparison across more time series since every single time series has a different impact on the overall results [5]. For example, it has been shown that five of the 1001 series from the M-competition dominated the RMSE ranking of the forecasting methods and the remaining 996 series had only little impact on the ranking [7].

For a comparison of distinct methods across more time series, scale-independent measures have to be used. *Symmetric Mean Absolute Percentage Error* (SMAPE) and *Mean Absolute Scaled Error* (MASE) [61], that are among the most suggested ones, are given as follows:

$$\text{SMAPE} = \frac{1}{h}\sum_{t=T+1}^{T+h} \frac{|y_t - \hat{y}_t|}{\frac{(|y_t|+|\hat{y}_t|)}{2}} \times 100\%, \tag{1.1.6}$$

$$\text{MASE} = \frac{1}{h}\sum_{t=T+1}^{T+h} \frac{|e_t|}{\frac{1}{T-1}\sum_{j=2}^{T} |y_j - y_{j-1}|}. \tag{1.1.7}$$

Although the SMAPE was originally proposed in [4] in a different form, formula (1.1.6) adopts the variant used in [3] since it does not lead to negative values (ranging from 0% to 200%). However, there are still two main SMAPE drawbacks [61]: the denominator may be close to zero, and a heavier penalty is given to under-forecasting when compared to over-forecasting.

More recently proposed [61], the MASE is more widely applicable and does not hold the SMAPE disadvantages. When MASE > 1, the forecasts are worse (on average) when compared with the in-sample one-step forecasts of the naïve random-walk method. In other words, the MASE compares the average out-sample error with the average in-sample first difference and it relativizes the prediction error with respect to fluctuations from the past.

Generally, it is not suggested to rely only on one error measure [5, 30] since distinct results may be obtained for different measures. However, it is worth recalling the empirical evidence [30] that very good methods perform consistently well across multiple measures.

1.2 Introduction to fuzzy set theory

Let us briefly recall the basic elements of the fuzzy set theory which development has been initiated by L.A. Zadeh [128].

1.2.1 Fuzzy sets

A classical set A can be characterized by its *characteristic function* $\chi_A : U \to \{0, 1\}$

$$\chi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

Fuzzy set is a generalization of a classical set which allows elements to belong to it up to some extent that is given by the so-called *membership degree*. Usually,

membership degrees can take values from the whole interval $[0, 1]$. Characteristic function is then replaced by its generalization that is called *membership function* $\mu_A : U \to [0, 1]$. However, this notion and mainly to denotation becomes a bit redundant as we unify a fuzzy set A with its membership function μ_A and consider $A : U \to [0, 1]$ denoted by $A \underset{\sim}{\subseteq} U$. $\mathcal{F}(U)$ denotes the set of all fuzzy sets on U. The value $A(x)$ for $x \in U$ is called the *membership degree* of x to A. Immediately, it is clear that each classical set is a special case of a fuzzy set and therefore, fuzzy sets do not contradict classical sets but generalize them.

The definition of a fuzzy set allows us to model vague human language notions. Especially, so-called evaluative linguistic expressions (*small, very big, more or less medium, about five, etc.*) [87, 89] can be successfully modelled by fuzzy sets, see Figure 1.3.

Definition 2 Let $A \underset{\sim}{\subseteq} U$. Then the *support* of A and the *kernel* of A are the following sets

$$\mathrm{Supp}(A) = \{x \mid x \in U,\ A(x) > 0\}, \tag{1.2.1}$$

$$\mathrm{Ker}(A) = \{x \mid x \in U,\ A(x) = 1\}, \tag{1.2.2}$$

respectively.

The support is a set of elements from a domain with a non-zero membership degree to a given fuzzy set or a set of those elements of U which at least partially belong to a given fuzzy set. The kernel is a set of elements of U which fully belong to a given fuzzy set.

If the universe U is a linear subspace of real numbers $\mathbb{R}$, we can define convexity of a fuzzy set.

Definition 3 Let $A \underset{\sim}{\subseteq} U$ and U be a linear subspace of $\mathbb{R}$. A is called *convex* if for

any $x, y \in U$ and for any $\lambda \in [0,1]$ the following formula holds

$$A(\lambda x + (1-\lambda)y) \geq A(x) \wedge A(y). \tag{1.2.3}$$

1.2.2 T-norms, t-conorms and residual (bi)implications

The main idea for introducing the *triangular norms* (t-norms) was to generalize the concept of the triangular inequality. T-norms serve as natural generalizations of classical conjunctions. We recall only some basic definitions and facts and refer to very rich literature [72, 95, 49].

Definition 4 A binary operation $* : [0,1]^2 \to [0,1]$ is called *triangular norm* (t-norm) if it fulfills the following properties for all $x, y, z \in [0,1]$:

$$\begin{aligned} x * y &= y * x && \text{(commutativity)},\\ x * (y * z) &= (x * y) * z && \text{(associativity)},\\ x \leq y &\Longrightarrow x * z \leq y * z && \text{(monotonicity)},\\ x * 1 &= x && \text{(boundary condition)}. \end{aligned}$$

Example 1 *Below, we show the most known examples of continuous t-norms which serve as natural interpretations of a generalized conjunction:*

(1) *Minimum t-norm* $x * y = x \wedge y$,

(2) *Product t-norm* $x \odot y = x \cdot y$,

(3) *Łukasiewicz t-norm* $x \otimes y = \max(0, x + y - 1)$.

Another operation associated with the t-norm is called *triangular conorm* (t-conorm) and it corresponds (due to its behavior) to a generalization of the classical disjunction. Consequently, it serves for the interpretation of unions of fuzzy sets.

Definition 5 A t-conorm is a binary operation $\sqcup : [0,1]^2 \to [0,1]$ which has the properties of commutativity, associativity and monotonicity (introduced in Definition 4) and fulfills the following boundary condition for all $x \in [0,1]$:

$$0 \sqcup x = x.$$

A t-conorm dual to a given t-norm $*$ is given by

$$a \sqcup b = 1 - (1-a) * (1-b).$$

Let us recall that for each t-norm, there exists a dual t-conorm and vice-versa. Particularly, let $*$ be a t-norm. Then the binary operation $\sqcup$ on $[0,1]$ given as follows

$$a \sqcup b = 1 - ((1-a) * (1-b))$$

is a t-conorm that is dual to the t-norm $*$.

Example 2 *The most important t-conorms dual to the t-norms from Example 1 are:*

(1) *Maximum t-conorm (dual to minimum t-norm)* $x \sqcup y = x \vee y$,

(2) *Product t-conorm (dual to product t-norm)* $x \sqcup y = x + y - x \cdot y$,

(3) *Łukasiewicz t-conorm (dual to Łukasiewicz t-norm)* $x \oplus y = \min(1, x + y)$.

It follows from the definition of the t-norm that it is a monoidal operation on $[0,1]$. Furthermore, $\langle [0,1], \wedge, \vee \rangle$ is a complete lattice. Therefore, we can define the residuation operation in the following form.

Definition 6 Let $*$ be a t-norm. The *residuation* operation $\to_*$: $[0,1]^2 \to [0,1]$ is defined by

$$x \to_* y = max\{z \,|\, x * z \leq y\}. \tag{1.2.4}$$

The residuation operation serves as an operation representing the generalized implication for fuzzy logics.

Moreover, we will use the following derived operation

$$x \leftrightarrow_* y = (x \rightarrow_* y) \wedge (y \rightarrow_* x)$$

that will be called *biresiduation* or a biresiduum. Biresiduation serves as a representation of an appropriate multi-valued biimplication, i.e. a fuzzy equivalence.

1.2.3 Operations on fuzzy sets, fuzzy relations

In this subsection, we recall elementary definitions of operations on fuzzy sets.

Definition 7 Let $A, B \underset{\sim}{\subseteq} U$. Then the *intersection* $C = A \cap B$ and the *union* $C = A \cup B$ of these two fuzzy sets is a fuzzy set $C \underset{\sim}{\subseteq} U$ given as follows

$$C(x) = A(x) \wedge B(x), \tag{1.2.5}$$

$$C(x) = A(x) \vee B(x), \tag{1.2.6}$$

respectively.

Besides original Definitions 7, the $*$-intersection $\sqcup$-union can be defined by t-norms and t-conorms, respectively. Given a t-norm $*$ and a t-conorm $\sqcup$, then the $*$-intersection $C = A \cap_* B$ and the $\sqcup$-union $C = A \cup_\sqcup B$ are given by

$$C(x) = A(x) * B(x), \tag{1.2.7}$$

$$C(x) = A(x) \sqcup B(x), \tag{1.2.8}$$

respectively.

Similarly to the classical set theory, also in the fuzzy set theory we may construct relations and Cartesian products. Naturally, a fuzzy relation is a fuzzy set defined on a Cartesian product of universes.

Definition 8 An *n-ary fuzzy relation* R is a fuzzy set on a Cartesian product $U_1 \times \cdots \times U_n$ of n universes.

The membership degree $R(x_1, \ldots, x_n)$ expresses the degree, in which the n-tuple $(x_1, \ldots, x_n)$ is in the fuzzy relation R.

Fuzzy relations play an important role in fuzzy modeling, more specifically, they interpret so-called *fuzzy rule bases*.

Fuzzy rule bases are sets of *fuzzy rules* given in a natural language:

$$\begin{aligned} \mathcal{R}_1 &:= \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_1\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_1 \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \mathcal{R}_m &:= \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_m\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_m \end{aligned} \tag{1.2.9}$$

that intuitively express a relationship between X and Y which may model a given problem (control law, decision-making strategy etc). *Linguistic expressions* [130, 132, 133, 92] $\mathcal{A}_i, \mathcal{B}_i$ are modelled by fuzzy sets A_i, B_i taking values from universes U and V, respectively. Fuzzy rule $\mathcal{R}_i$ is modelled by a fuzzy relation $R_i \subsetsim U \times V$ and the whole fuzzy rule base by a fuzzy relation $R \subsetsim U \times V$.

In case of a crisp input $x' \in U$, the inferred output obtained with help of the fuzzy rule base modelled by a fuzzy relation R, is a fuzzy set B on V given by

$$B(y) = R(x', y), \quad \text{for any } y \in V.$$

There are two standard approaches to model fuzzy rule base (1.2.9) by a fuzzy relation. The first one respects the conditional nature of the rules and employs a (usually residuated) fuzzy implication $\to_*$. In such case, the fuzzy relation is constructed as follows:

$$R(x, y) = \bigwedge_{i=1}^{m} \left(A_i(x) \to_* B_i(y) \right),$$

i.e., as a conjunction of implicative rules modelled by

$$R_i(x, y) = A_i(x) \to_* B_i(y).$$

The second approach, that is more often applied in real world applications, stems from the seminal work od Mamdani and Assilian [83]. It models fuzzy rule base (1.2.9) by the following fuzzy relation

$$R(x, y) = \bigvee_{i=1}^{m} \left(A_i(x) * B_i(y)\right),$$

which obviously does not implement any kind of implication and ignores the conditional nature of fuzzy IF-THEN rules.

In this thesis, we will use another approach that is described in Section 1.3. This approach is closer to the implicative one because it also views the individual rules as implicative ones and models them with help of the Łukasiewicz residual implication. However, it directly involves the theory of evaluative linguistic expressions. Moreover, it does not model the whole rule base by a conjunction of fuzzy relations modelling the individual rules.

1.3 Perception-based Logical Deduction

1.3.1 Evaluative linguistic expressions

One of main constituents of systems of fuzzy/linguistic IF-THEN rules are *evaluative linguistic expressions* [89], in short *evaluative expressions*. They are special expressions of natural language that are used whenever it is important to evaluate a decision situation, to specify the course of development of some process, and in many other situations. Typical examples of evaluative linguistic expressions are expressions *very large, extremely expensive, more or less hot*, etc. Note that their

importance and the potential to model their meaning mathematically have been pointed out by L. A. Zadeh (e.g., in [129, 131, 134]).

A simple form of evaluative expressions keeps the following structure:

$$\langle \text{linguistic hedge} \rangle \langle \text{atomic evaluative expression} \rangle \tag{1.3.1}$$

Atomic evaluative expressions comprise any of the *canonical* adjectives *small, medium, big*, abbreviated in the following as Sm, Me, Bi, respectively. It is important to stress that these words are in practice often replaced by other kinds of evaluative words, such as "thin", "old", "new", etc., depending on the context of speech. Let us note that in many situations, it is advantageous to extend the set of atomic evaluative expressions by the evaluative expression *zero*. Moreover, a special expression `any`, is also introduced.

Linguistic hedges are specific adverbs that make the meaning of the atomic expression more or less precise. We may distinguish hedges with *narrowing effect*, e.g. *very*, *extremely*, etc. and with *widening effect*, e.g. *roughly*, *more or less*, etc. In general, there can be a finite number of hedges with narrowing effect and a finite number of hedges with widening effect. In practical applications, their number is of course limited. In the following text, we, without loss of generality, use the hedges introduced in Table 1.1 that were successfully used in real applications [122] and that are implemented in the LFLC software package [35]. As a special case, the ⟨linguistic hedge⟩ can be empty. This enables us to include atomic evaluative expressions into the class of simple ones and develop a unified theory of their meaning. Note that our hedges are of so-called inclusive type [33], which means that extensions of more specific evaluative expressions are included in less specific ones[†)], see Figure 1.1.

[†)]The expression `any` is modelled by a fuzzy set that attains normality at all points of a given universe.

Narrowing effect	Widening effect
very (Ve)	more or less (ML)
significantly (Si)	roughly (Ro)
extremely (Ex)	quite roughly (QR)

Table 1.1: Linguistic hedges and their abbreviations.

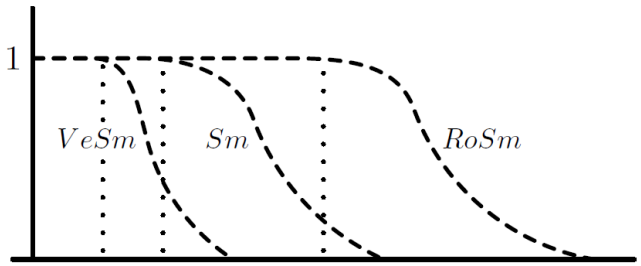


Figure 1.1: Graphical presentation of extensions (fuzzy sets) that interpret linguistic expressions *very small*, *small* and *roughly small*.

Evaluative expressions of the form (1.3.1) will generally be denoted by script letters $\mathcal{A}, \mathcal{B}$, etc. They are used to evaluate values of some variable X. The resulting expressions are called *evaluative linguistic predications*, and have the form

$$X \text{ is } \mathcal{A}.$$

Examples of evaluative predications are "temperature is very high", "price is low", etc.

The model of the meaning of evaluative expressions and predications makes distinction between *intensions* and *extensions* in various *contexts*. The context characterizes a range of possible values. This range can be characterized by a triple of numbers $\langle v_L, v_M, v_R \rangle$, where $v_L, v_M, v_R \in \mathbb{R}$ and $v_L < v_M < v_R$. These numbers characterize the minimal, middle, and maximal values, respectively, of the evaluated characteristics in the specified context of use. Let us stress that the middle value

v_M is not required to be in the exact center of the interval $[v_L, v_R]$. Therefore, we will identify the notion of context with the triple $\langle v_L, v_M, v_R \rangle$. By $u \in w$ we mean $u \in [v_L, v_R]$. In the sequel, we will work with a set of contexts W

$$W \subset \{\langle v_L, v_M, v_R \rangle \mid v_L, v_M, v_R \in \mathbb{R}, v_L < v_M < v_R\}$$

that are given in advance.

The *intension* of an evaluative predication "X is $\mathcal{A}$" is a certain formula whose interpretation is a function

$$\mathrm{Int}(X \text{ is } \mathcal{A}) : W \longrightarrow \mathcal{F}(\mathbb{R}), \tag{1.3.2}$$

i.e., it is a function that assigns a fuzzy set to any context from the set W. More details and explanations can be found, e.g., in [89].

Given an intension (1.3.2) and a context $w \in W$, we can define the *extension* of "X is $\mathcal{A}$" in the context w as a fuzzy set

$$\mathrm{Int}(X \text{ is } \mathcal{A})(w) \subsetsim [v_L, v_R],$$

where $\subsetsim$ denotes the relation of fuzzy subsethood.

We extend the theory of evaluative linguistic expressions by the following *partition axiom*: There exists no context $w \in W$ in which there would exists a $u_0 \in w$ such that

$$(\mathrm{Int}(X \text{ is } \mathcal{A})(w))(u_0) = (\mathrm{Int}(X \text{ is } \mathcal{B})(w))(u_0) = 1 \tag{1.3.3}$$

for $\mathcal{A}, \mathcal{B}$ with different atomic evaluative expressions. Partition axiom brings an additional assumption to the theory that extensions of evaluative linguistic expressions cannot overlap in the degree one if these expression are not of the same atomic type. Indeed, no element u_0 in any world is naturally assumed to belong in the degree one to a fuzzy set of small objects as well as of medium or big objects – no matter the influence of the widening or narrowing effect of applied linguistic hedges.

Convention 1 *For the sake of brevity and simplicity and having in mind that an extension is a fuzzy set on a given context, we will omit the notion of extension from our consideration when appropriate and write only the abbreviated forms:*

$$A := (\mathrm{Int}(X \textit{ is } \mathcal{A})(w)), \quad w \in W, \textit{ and}$$
$$A(u_0) := (\mathrm{Int}(X \textit{ is } \mathcal{A})(w))(u_0), \quad u_0 \in w$$

if there is no danger of any confusion caused by the fact that the left-hand side does not explicitly mention the chosen context w and variable X.

To be able to state relationships among evaluative expressions, for example, when one expression "covers" another, we need an ordering relation defined on the set of them. Let us start with the ordering on the set of linguistic hedges. We may define the ordering $\leq_{\mathrm{H}}$ of examples of hedges introduced in Table 1.1 as follows:

$$\mathrm{Ex} \leq_{\mathrm{H}} \mathrm{Si} \leq_{\mathrm{H}} \mathrm{Ve} \leq_{\mathrm{H}} \langle\mathrm{empty}\rangle \leq_{\mathrm{H}} \mathrm{ML} \leq_{\mathrm{H}} \mathrm{Ro} \leq_{\mathrm{H}} \mathrm{QR}.$$

We extend the theory of evaluative linguistic expressions by the following *inclusion axiom*. Let $\mathrm{Ker}(A)$ denotes the kernel of a fuzzy set A. For any w,

$$\mathrm{Int}(X \text{ is } \langle\mathrm{hedge}\rangle_i\mathcal{A})(w) \subseteq \mathrm{Int}(X \text{ is } \langle\mathrm{hedge}\rangle_j\mathcal{A})(w) \tag{1.3.4}$$

and

$$\mathrm{Ker}(\mathrm{Int}(X \text{ is } \langle\mathrm{hedge}\rangle_i\mathcal{A})(w)) \subset \mathrm{Ker}(\mathrm{Int}(X \text{ is } \langle\mathrm{hedge}\rangle_j\mathcal{A})(w)) \tag{1.3.5}$$

hold for any atomic expression $\mathcal{A}$ under the assumptions $\langle\mathrm{hedge}\rangle_i \leq_{\mathrm{H}} \langle\mathrm{hedge}\rangle_j$, $i \neq j$.

Based on $\leq_{\mathrm{H}}$ we may define an ordering $\leq_{\mathrm{LE}}$ of evaluative expressions.

Definition 9 Let $\mathcal{A}_i$, $\mathcal{A}_j$ be two evaluative expressions such that $\mathcal{A}_i := \langle\mathrm{hedge}\rangle_i\mathcal{A}$ and $\mathcal{A}_j := \langle\mathrm{hedge}\rangle_j\mathcal{A}$. Then we write

$$\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$$

if $\mathcal{A} \in \{\mathrm{Sm}, \mathrm{Me}, \mathrm{Bi}\}$ and $\langle \text{hedge} \rangle_i \leq_{\mathrm{H}} \langle \text{hedge} \rangle_j$. Moreover,

$$\mathcal{A}_i \leq_{\mathrm{LE}} \mathsf{any}$$

for arbitrary $\mathcal{A}_i$.

In other words, evaluative expressions of the same type are ordered according to their specificity which is given by the hedges appearing in the expressions. If we are given two evaluative expressions with different atomic expressions, we cannot order them by $\leq_{\mathrm{LE}}$. There is an obvious natural extension of $\leq_{\mathrm{LE}}$ to the ordering of evaluative predications of the form (X is $\mathcal{A}$).

1.3.2 Fuzzy/linguistic IF-THEN rules and linguistic descriptions

Evaluative predications occur in conditional clauses of natural language of the form

$$\mathcal{R} := \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B} \tag{1.3.6}$$

where $\mathcal{A}, \mathcal{B}$ are evaluative expressions. The linguistic predication "X is $\mathcal{A}$" is called the *antecedent* and "Y is $\mathcal{B}$" is called the *consequent* of rule (1.3.6). Of course, the antecedent may consist of more evaluative predications, joined by the connective "AND". The clauses (1.3.6) will be called fuzzy/linguistic IF-THEN rules in the sequel.

Fuzzy/linguistic IF-THEN rules are gathered in a fuzzy rule base. But in order to distinguish between fuzzy rule bases that are modelled by one of the standard fuzzy relational approaches and between fuzzy rules that directly employ the theory of evaluative linguistic expressions and thus, are supposed to be modelled in a different

way, we introduce the notion of a *linguistic description* $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$, $m \geq 1$:

$$\begin{aligned} \mathcal{R}_1 &:= \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_1\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_1, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \mathcal{R}_m &:= \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_m\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_m. \end{aligned} \tag{1.3.7}$$

Because each rule in (1.3.7) is taken as a specific *conditional sentence of natural language*, a linguistic description can be understood as a *specific kind of a (structured) text*. This text can be viewed as a *model* of specific behavior of the system in concern.

The *intension of a fuzzy/linguistic IF-THEN rule* $\mathcal{R}$ in (1.3.6) is a function

$$\mathrm{Int}(\mathcal{R}) : W \times W \longrightarrow \mathcal{F}(\mathbb{R} \times \mathbb{R}). \tag{1.3.8}$$

This function assigns to each context $w \in W$ and each context $w' \in W$ a *fuzzy relation* in $w \times w'$. The latter is an *extension* of (1.3.8).

We also need to consider a linguistic phenomenon of topic-focus articulation (see [52, 111]), which in the case of linguistic descriptions requires us to distinguish the following two sets:

$$\begin{aligned} \mathit{Topic}_{LD} &= \{\mathrm{Int}(X \text{ is } \mathcal{A}_j) \mid j = 1, \ldots, m\}, \\ \mathit{Focus}_{LD} &= \{\mathrm{Int}(Y \text{ is } \mathcal{B}_j) \mid j = 1, \ldots, m\}. \end{aligned}$$

The phenomenon of topic-focus articulation plays an important role in the inference method called perception-based logical deduction described below.

Convention 2 *Besides the above introduced notions of topic and focus, it is sometimes advantageous to introduced the following notation:*

$$\mathit{Topic}^{w}_{LD} = \{\mathrm{Int}(X \textit{ is } \mathcal{A}_j)(w) \mid j = 1, \ldots, m\}$$

that will denote the set of extensions of evaluative predications that are contained in $Topic_{LD}$ *knowing the particular context* w*. This notation will be used later on when defining the function of local perception. In the view of Convention 1 one can also easily introduce the* $Topic^w_{LD}$ *as follows:*

$$Topic^w_{LD} = \{A_j \mid j = 1, \ldots, m\}.$$

Finally, we define the ordering of extensions with respect to an observation.

Definition 10 Let us be given a linguistic description LD (1.3.7), a context $w \in W$, an observation $u_0 \in w$ and two extensions A_i and A_j from the $Topic^w_{LD}$. We write

$$A_i \leq_{(u_0,w)} A_j$$

either if

$$A_i(u_0) > A_j(u_0)$$

or if

$$A_i(u_0) = A_j(u_0), \quad \text{and } \mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j.$$

It should be noted that usually the antecedents of a linguistic description contain evaluative predications which are formed by a conjunction of more than one evaluative predication. In other words, we usually meet the following situation

$$(X \text{ is } \mathcal{A}_i) := (X_1 \text{ is } \mathcal{A}_{i1}) \ \mathsf{AND} \ \cdots \ \mathsf{AND} \ (X_K \text{ is } \mathcal{A}_{iK}).$$

In this case, the extended ordering $\leq_{\mathrm{LE}}$ of compound evaluative expressions is preserved with respect to the components:

$$\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j \quad \text{if} \quad \mathcal{A}_{ik} \leq_{\mathrm{LE}} \mathcal{A}_{jk} \quad \text{for all } k = 1, \ldots, K. \tag{1.3.9}$$

The extension of the compound linguistic predication is given as follows

$$(\mathrm{Int}(X \text{ is } \mathcal{A}_i)(w_1, \ldots, w_K))(u_1, \ldots, u_K) = \bigwedge_{k=1}^{K} (\mathrm{Int}(X_k \text{ is } \mathcal{A}_{ik})(w_k))(u_k) \tag{1.3.10}$$

or in a shorter notation based on Convention 1 as follows

$$A_i(u_1, \ldots, u_K) = \bigwedge_{k=1}^{K} A_{ik}(u_k). \tag{1.3.11}$$

Then, the final ordering $\leq_{(u_0,w)}$ is analogous to the one-dimensional one. First, we determine the membership degrees of $u_0 = (u_{01}, \ldots, u_{0K})$ in extensions of compound predications and in case of equal values, we determine the ordering $\leq_{\text{LE}}$ of the expressions appearing in the components of the compound ones as given by (1.3.9).

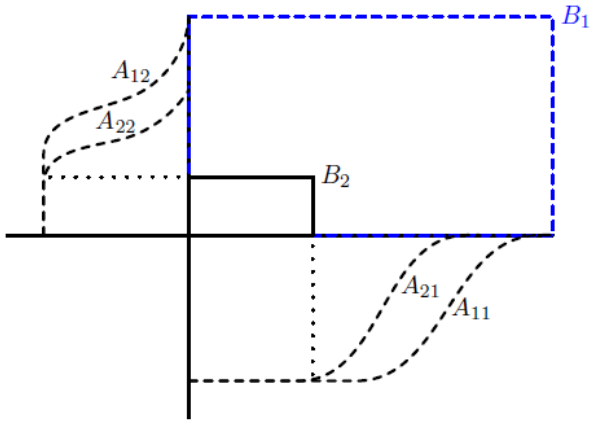


Figure 1.2: Visualization of fuzzy rules $\mathcal{R}_1, \mathcal{R}_2$. Displayed rectangles symbolically denote areas covered by antecedents of given fuzzy rules with their respective consequents $\mathcal{B}_1, \mathcal{B}_2$.

On Figure 1.2, we provide the readers with a visualization of two fuzzy rules with two input variables. This double-input example is fully sufficient for intuitive understanding. Hence, it will be used for an illustration of distinct situations in the rest of this thesis. Note that the rectangles denote areas covered by antecedents of given fuzzy rules, in other words, they delimit areas where the antecedent of the given rule is minimal with respect to the ordering $\leq_{(u_0,w)}$. Each rectangle is also denoted by a respective consequent (B_1 or B_2).

1.3.3 Perception-based logical deduction

The perception-based logical deduction (PbLD) is a special inference method aimed at the derivation of results based on fuzzy/linguistic IF-THEN rules. A perception is understood as an evaluative expression assigned to the given input value in the given context. The choice of perception is not arbitrary. It depends on the specified linguistic description, too. In other words, perception is always chosen from evaluative expressions which occur in antecedents of fuzzy/linguistic IF-THEN rules. For details, see [88, 90, 122]. The perception is determined by a special function called the *local perception* which is based on the ordering $\leq_{(u_0,w)}$.

Definition 11 Let LD be a linguistic description (1.3.7). The *local perception* function is a mapping $LPerc^{LD} : w \times W^K \longrightarrow \mathcal{P}(Topic^w_{LD})$ assigning to each value $u_0 = (u_1, \ldots, u_K) \in w$ for $w = (w_1, \ldots, w_K) \in W^K$ a subset of extensions minimal with respect to the ordering $\leq_{(u_0,w)}$

$$LPerc^{LD}(u_0, w) = \{A_i \mid A_i(u_0) > 0 \;\&\; \forall A_j \in Topic^w_{LD} : (A_j \leq_{(u_0,w)} A_i) \Rightarrow (\mathcal{A}_j = \mathcal{A}_i)\} \quad (1.3.12)$$

The local perception function has a key role in the definition of the PbLD inference mechanism. It can be viewed as a function that"fires" chosen rules.

Definition 12 Let LD be a linguistic description (1.3.7). Let us consider a context $w \in W$ for the variable X and a context $w' \in W$ for Y. Let an observation $X = u_0$ in the context w be given, where $u_0 \in w$. Then, the following *rule of perception-based logical deduction* can be introduced:

$$r_{PbLD} : \frac{LPerc^{LD}(u_0, w), LD}{C} \quad (1.3.13)$$

where

$$C = \bigcap \{C_{i_\ell} \mid C_{i_\ell}(v) = A_{i_\ell}(u_0) \to B_{i_\ell}(v) \ \& \ A_{i_\ell} \in LPerc^{LD}(u_0, w)\}, \ v \in w',$$

where $\to$ is the Łukasiewicz implication and $\bigcap$ is the Gödel intersection.

Informally, C is the conclusion corresponding to the observation in the following way. Inputs to this inference rule are the linguistic description LD and the local perception $LPerc^{LD}(u_0, w)$ given by (1.3.12). This local perception is formed by a set of fuzzy sets A_{i_ℓ}, $\ell = 1, \ldots, L$, that are chosen from the topic of LD according to (1.3.12). Formula (1.3.12) chooses these antecedents, which best fit the given numerical input u_0 and thus, they should be fired. Then, the individual conclusions C_{i_ℓ} contained in C are computed as $A_{i_\ell}(u_0) \to B_{i_\ell}(v)$ for all $v \in w'$. It means that for each $A_{i_\ell} \in LPerc^{LD}(u_0, w)$ we take the i_ℓ-th IF-THEN rule from LD and compute the conclusion, for the time being forgetting other IF-THEN rules from LD.

Remark 1 *Let us recall that the final inference output is aggregated using the intersection of all elements in C. Thus, it is easy to see that whenever an LD contains two rules:*

$$\mathcal{R}_i := \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_i\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_i,$$
$$\mathcal{R}_j := \ \mathsf{IF}\ X\ \mathsf{is}\ \mathcal{A}_j\ \mathsf{THEN}\ Y\ \mathsf{is}\ \mathcal{B}_j,$$

such that $\mathcal{A}_i = \mathcal{A}_j$ and $\mathcal{B}_i \leq_{\mathrm{LE}} \mathcal{B}_j$, the rule $\mathcal{R}_j$ is trivially redundant.

This fact may be used in the preprocessing of linguistic descriptions in order to efficiently decrease the number of investigated rules.

The idea of assigning local perceptions is not restricted only to the topic of a given linguistic description. If we generalize it slightly, we can learn the linguistic description on the basis of the given data. More details about this method can be

found in [20]. Let us remark that we have successfully implemented this method in the software system LFLC (see [35]).

Lemma 1.3.1

Let LD be a linguistic description (1.3.7) and let $\mathcal{R}_i \in LD$. For each $w \in W$ there exists $u_0 \in w$ such that

$$LPerc^{LD}(u_0, w) = \{A_i\}. \tag{1.3.14}$$

PROOF: Suppose first that $K = 1$ (there is just one antecedent variable). Suppose that the following condition (K) holds: for every i, there is $u_0 \in w$ such that $A_i(u_0) = 1$, and for any $j \in 1, \ldots, m$, if $A_j(u_0) = 1$, then $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$. It is easy to see that the above condition is a sufficient condition for (1.3.14) to hold: From (1.3.12) and the definition of $\leq_{(u_0,w)}$ it follows that $A_i \in LPerc^{LD}(u_0, w)$. Suppose that $A_j \in LPerc^{LD}(u_0, w)$, too. According to the definition of $\leq_{(u_0,w)}$, it is possible only if $A_j(u_0) = 1$. But then either $\mathcal{A}_i <_{LE} \mathcal{A}_j$ and $A_j \notin LPerc^{LD}(u_0, w)$ (contradiction), or $\mathcal{A}_i = \mathcal{A}_j$ from (K).

The condition (K) for evaluative linguistic expressions easily follows from the partition axiom (1.3.3) and the inclusion axiom (1.3.5). First, from (1.3.5) it follows that there is $u_0 \in w$ such that $A_i(u_0) = 1$ and that $A_j(u_0) < 1$ for all expressions $\mathcal{A}_j$ such that $\mathcal{A}_j \leq_{\mathrm{LE}} \mathcal{A}_i$. Axiom (1.3.3) implies that there is no evaluative expression $\mathcal{A}_k$ with different atomic expression than that of $\mathcal{A}_i$ such that $A_k(u_0) = 1$.

Without a loss of generality,suppose that $K = 2$. Then,

$$(X \text{ is } \mathcal{A}_i) = (X_1 \text{ is } \mathcal{A}_{i_1}) \ \mathsf{AND} \ (X_2 \text{ is } \mathcal{A}_{i_2}).$$

From partition and inclusion axioms it follows that there exist $u_{01} \in w_1$ and $u_{02} \in w_2$ for which (K) hold. From the definition of $\leq_{\mathrm{LE}}$ (1.3.9) for more than one antecedent variable and the definition of the extension of a compound linguistic predication

(1.3.11) it follows that (K) holds for $u_0 = (u_{01}, u_{02})$, too, and the proof is finished.

□

1.3.4 Defuzzification

Sometimes, for example in fuzzy control or in other fully automatized systems, it is necessary to work with the accurate values. Therefore, we have to introduce the concept of *defuzzification*. Defuzzification assigns an element of a given universe to a fuzzy set on the universe in such a way that the assigned element appropriately represents the fuzzy set. Thus, it can be successfully employed at the end of the inference process to obtain a crisp representant (e.g. a control action) of the inferred output fuzzy set. Mathematically, defuzzification is a mapping $\mathrm{DEF} : (\mathcal{F}(U) \setminus \{\emptyset\}) \to U$ such that

$$\mathrm{DEF}(A) \in \mathrm{Supp}(A)$$

holds for any nonempty fuzzy set $A \in \mathcal{F}(U)$.

Now, we will work only with fuzzy sets having a finite support, i.e., the fuzzy sets have the form

$$A = \{^{a_1}/u_1, \ldots, ^{a_r}/u_r\}. \tag{1.3.15}$$

This restriction is motivated by practical reasons. Note that most of the formulas given below can be defined for fuzzy sets on infinite universes as well.

The most commonly used defuzzification method is the *Center of Gravity* (COG). This is mainly due to the fact that it is an appropriate defuzzification when the Mamdani-Assilian models of fuzzy rule bases are used.

Definition 13 Let $A \subsetneq U$. Then the *center of gravity* of A is defined as

$$\mathrm{COG}(A) = \frac{\sum_{k=1}^{r} A(u_k) \cdot u_k}{\sum_{k=1}^{r} A(u_k)}.$$

Another useful defuzzification methods are maxima methods. These methods consider values with the maximum membership.

Definition 14 Let $A \subsetneq U$. Then the *mean of maxima* of A is defined as

$$\mathrm{MOM}(A) = \frac{1}{r_{max}} \sum_{j=1}^{r_{max}} u_j^{max}.$$

where $u_j^{max} = u_j$, if $A(u_j) = max\{A(u_k)|k = 1, \ldots, r\}$, that is, $\{u_j^{max}|j = 1, \ldots, r_{max}\}$ are all the elements of the support of fuzzy set (1.3.15) whose membership degree $A(u_j^{max})$ is maximal.

The mean of maxima method is an appropriate defuzzification when implicative models of fuzzy rule bases are used.

Definition 15 Let $A \subsetneq U$. Then the *first of maxima* of A and the *last of maxima* of A are defined as

$$\mathrm{FOM}(A) = \min\{u_j^{max}|j = 1, \ldots, r_{max}\},$$
$$\mathrm{LOM}(A) = \max\{u_j^{max}|j = 1, \ldots, r_{max}\},$$

where $\{u_j^{max}|j = 1, \ldots, r_{max}\}$ are all the elements of the support of fuzzy set (1.3.15) whose membership degree $A(u_j^{max})$ is maximal.

Defuzzifications FOM and LOM are not usually directly implemented although they played an important role in case of use of specific implicative models of monotone fuzzy rule bases [121]. Their importance for this thesis is given by the fact that they are used in the construction of the *Defuzzification of Evaluative Expressions* (DEE) that has been designed specifically for the outputs of the PbLD inference mechanism. In principle, this defuzzification is a combination of FOM, MOM and LOM that are applied based on the classification of the output fuzzy sets, see [87, 92].

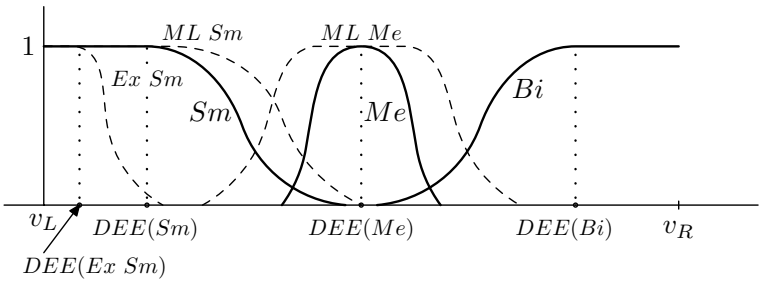


Figure 1.3: Fuzzy sets that model extensions of some expressions and their defuzzifications.

Definition 16 Let $A \subsetneq U$. Then the *defuzzification of evaluative expressions* of A is defined as

$$\mathrm{DEE}(A) = \begin{cases} \mathrm{LOM}(A) & \text{if} \quad A \text{ is non-increasing,} \\ \mathrm{FOM}(A) & \text{if} \quad A \text{ is non-decreasing,} \\ \mathrm{MOM}(A) & \text{otherwise.} \end{cases}$$

Results of the DEE defuzzification are depicted on Figure 1.3.

1.4 Fuzzy transform

The fuzzy transform is a special technique that can be applied to a continuous function defined on a fixed real interval $[a, b] \subset \mathbb{R}$. The essential idea is to transform a given function defined in one space into another, which is usually a simpler space, and then to transform it back. The simpler space consists of a finite vector of numbers obtained on the basis of the well-established *fuzzy partition* of the domain of the given function. The inverse transform then leads to a function approximately reconstructing the original one. Thus, the first step, which is sometimes called the *direct fuzzy transform*, results in a vector of averaged functional values. The

second step, which is called the *inverse transform*, converts this vector into another continuous function, which approximates the original one. In this section, we provide a brief overview of the main concepts. More details can be found in [100].

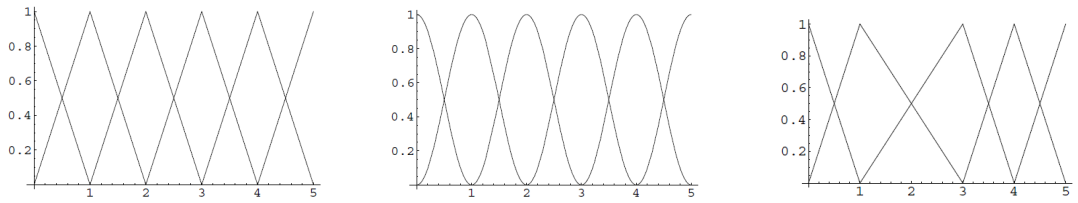
The fuzzy transform is defined with respect to a *fuzzy partition*, which consists of *basic functions*.

Definition 17 Let $c_1 < \cdots < c_n$ be fixed nodes within $[a, b]$, such that $c_1 = a, c_n = b$ and $n \geq 2$. We say that fuzzy sets $A_1, \ldots, A_n \in \mathcal{F}([a, b])$ are *basic functions* forming a fuzzy partition of $[a, b]$ if they fulfill the following conditions for $i = 1, \ldots, n$:

1. $A_i(c_i) = 1$;
2. $A_i(x) = 0$ for $x \notin (c_{i-1}, c_{i+1})$, where for uniformity of notation, we put $c_0 = c_1 = a$ and $c_{n+1} = c_n = b$;
3. A_i is continuous;
4. A_i strictly increases on $[c_{i-1}, c_i]$ and strictly decreases on $[c_i, c_{i+1}]$;
5. for all $x \in [a, b]$,
$$\sum_{i=1}^{n} A_i(x) = 1. \tag{1.4.1}$$

Usually, the *uniform fuzzy partition* is considered (i.e., n equidistant nodes $c_i = c_{i-1} + h$, $i = 2, \ldots, n$ are fixed). Let us note that the shapes of the basic functions are not predetermined and can be chosen on the basis of additional requirements. For some examples of fuzzy partitions satisfying Definition 17, see Figure 1.4.

Definition 18 Let a fuzzy partition of $[a, b]$ be given by basic functions $A_1, \ldots, A_n, n \geq 2$, and let $f : [a, b] \longrightarrow \mathbb{R}$ be an arbitrary continuous function. The n-tuple of real numbers $\mathbf{F}[f] = (F_1[f], \ldots, F_n[f])$ given by
$$F_i[f] = \frac{\int_a^b f(x)A_i(x)dx}{\int_a^b A_i(x)dx}, \qquad i = 1, \ldots, n, \tag{1.4.2}$$



(a) Triangular uniform fuzzy partition

(b) Sinusoidal uniform fuzzy partition

(c) Triangular non-uniform fuzzy partition

Figure 1.4: Graphical presentation of several fuzzy partitions.

is a *direct fuzzy transform* of f with respect to the given fuzzy partition. The numbers $F_1[f], \ldots, F_n[f]$ are called the *components* of the fuzzy transform of f.

In practice, the function f is usually not given analytically, but we are at least provided with some data, obtained, for example, from measurements. In this case, Definition 18 can be modified in such a way that the definite integrals in Formula (1.4.2) are replaced by finite summations.

Let n basic functions forming a fuzzy partition of $[a, b]$ be given, and let the function f be given at $T > n$ fixed points $x_1, \ldots, x_T \in [a, b]$. We say that the set of points $\{x_1, \ldots, x_T\}$ is *sufficiently dense with respect to the fuzzy partition* if for every $i \in \{1, \ldots n\}$, there exists $t \in \{1, \ldots, T\}$ such that

$$A_i(x_t) > 0.$$

Definition 19 Let a fuzzy partition of $[a, b]$ be given by basic functions $A_1, \ldots, A_n, n \geq 2$, and let $f : [a, b] \longrightarrow \mathbb{R}$ be a function that is known on a set $\{x_1, \ldots, x_T\}$ of points that is sufficiently dense with respect to the given fuzzy partition. The n-tuple of real numbers $\mathbf{F}[f] = (F_1[f], \ldots, F_n[f])$ given by

$$F_i[f] = \frac{\sum_{t=1}^{T} f(x_t) A_i(x_t)}{\sum_{t=1}^{T} A_i(x_t)}, \qquad i = 1, \ldots, n, \tag{1.4.3}$$

is a *discrete direct fuzzy transform* of f with respect to the given fuzzy partition. The $F_1[f], \ldots, F_n[f]$ are the *components* of the (discrete) fuzzy transform of f.

It has been proven [100] that the components of the fuzzy transform are weighted mean values of the original function, where the weights are determined by the basic functions.

The original function f can be approximately reconstructed from $\mathbf{F}[f]$ using the following inversion formula.

Definition 20 Let $\mathbf{F}[f] = (F_1, \ldots, F_n)$ be the direct fuzzy transform of f with respect to $A_1, \ldots, A_n \in \mathcal{F}([a, b])$. Then the function $\hat{f}$ given on $[a, b]$ by

$$\hat{f}(x) = \sum_{i=1}^{n} F_i[f] A_i(x) \tag{1.4.4}$$

is called the *inverse fuzzy transform* of f.

The inverse fuzzy transform is a continuous function on $[a, b]$.

Let us recall the two main properties of the fuzzy transform. First, it should be stressed that for uniform fuzzy partitions, the sequence of the inverse fuzzy transform $\{\hat{f}\}_n$ uniformly converges to the original function f for $n \to \infty$ [100]. Assuming certain additional properties, an analogous result is valid even for non-uniform fuzzy partitions [123].

Second, the fuzzy transform components maintain a certain optimality, particularly by minimizing the *piecewise integral least square criterion.*

Consequently, the direct fuzzy transform may serve as a discrete approximate representation of a function and may be successfully used for numerical integration of a function, whereas, the inverse fuzzy transform is a suitable continuous approximation of a given function. For various properties of the fuzzy transform and detailed proofs, see [100, 102, 123].

1.5 Linguistic associations

In this thesis, we employ the so-called linguistic associations mining [2] for the fuzzy rule base identification. This approach, firstly introduced as GUHA method [48, 50], finds distinct statistically supported associations among attributes of given objects. Particularly, the GUHA method deals with Table 1.2 where $o_1, \ldots, o_n$ denote objects, $X_1, \ldots, X_m$ denote independent boolean attributes, Z denotes the dependent (explained) boolean attribute, and finally, symbols a_{ij} (or a_i) $\in \{0, 1\}$ denote whether an object o_i carries an attribute X_j (or Z) or not.

	X_1	X_2	$\ldots$	X_m	Z
o_1	a_{11}	a_{12}	$\ldots$	a_{1m}	a_1
o_2	a_{21}	a_{22}	$\ldots$	a_{2m}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
o_n	a_{n1}	a_{n2}	$\ldots$	a_{nm}	a_n

Table 1.2: Standard GUHA table.

The original GUHA allowed only boolean attributes to be involved, see [51]. Since most of the features of objects are measured on the real interval, standard approach assumed to booleanize the attributes by a partition of the interval into subintervals, see Example 3.

The goal of the GUHA method is to search for associations of the form

$$C(X_1, \ldots, X_p) \simeq D(Z) \tag{1.5.1}$$

where C, D are (compound) evaluative predications [89] containing only the connective `AND` and $X_1, \ldots, X_p$ for $p \leq m$ are all variables occurring in C. The C, D are called the *antecedent* and *consequent*, respectively. Generally, for the GUHA method, the well-known four-fold table is constructed, see Table 1.3.

Symbol a, in Table 1.3, denotes the number of positive occurrences of C as well as D; b is the number of positive occurrences of C and of negated D, i.e. of 'not D'.

	D	not D
C	a	b
not C	c	d

Table 1.3: Classical GUHA four-fold table.

Analogous meaning have the numbers c and d. For our purposes, only numbers a and b are important.

The relationship between the antecedent and consequent is described by so-called *quantifier* $\simeq$. There are many quantifiers that characterize validity of the association in the data [50]. For our task, we use the so-called a binary multitudinal quantifier $\simeq := \sqsubset_r^\gamma$. This quantifier is taken as true if

$$\frac{a}{a+b} > \gamma$$

and

$$\frac{a}{n} > r,$$

where $\gamma \in [0,1]$ is a confidence degree and $r \in [0,1]$ is a support degree.

Example 3 *For example, let us consider Table 1.4.*

	$\text{BMI}_{\leq 25}$	$\text{BMI}_{>25}$	$\text{Chol}_{>6.2}$	$\text{BP}_{>130/90}$
o_1	*1*	*0*	*0*	*0*
o_2	*0*	*1*	*1*	*1*
o_3	*0*	*1*	*0*	*1*
o_4	*1*	*0*	*0*	*0*
o_5	*0*	*1*	*1*	*1*
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
o_n	*0*	*0*	*1*	*1*

Table 1.4: Example of GUHA table. $\text{BMI}_{\leq 25}$ *denotes Body-Mass-Index lower or equal to 25,* $\text{BMI}_{>25}$ *denotes the same index above 25,* $\text{Chol}_{>6.2}$ *denotes Cholesterol higher than 6.2 and* $\text{BP}_{>130/90}$ *denotes Blood Pressure higher than 130/90. Objects* o_i *are particular patients.*

Depending on the chosen confidence and support degree, the GUHA method could

generate the following association:

$$C(\mathrm{BMI}_{>25}, \mathrm{Chol}_{>6.2}) \simeq D(\mathrm{BP}_{>130/90}) \ .$$

In many situations, including ours, the fuzzy variant of the GUHA method [73, 94] seems to be more appropriate. In the fuzzy variant of the method, the attributes are not boolean but rather vague (such as $\mathrm{BMI}_{\mathrm{ExBi}}$, $\mathrm{BMI}_{\mathrm{MLBi}}$, $\mathrm{Chol}_{\mathrm{VeBi}}$ etc.) and thus, values a_{ij}(or a_i) are elements of the interval $[0, 1]$ that express membership degrees, see Example 4.

The four-fold table analogous to Table 1.3 is constructed also for the fuzzy variant of the method. The difference is that the numbers a, b, c, d are not summations of 1s and 0s but summations of membership degrees of data into fuzzy sets representing the antecedent C and consequent D or their complements, respectively. Otherwise, the main idea of the method remains the same. Naturally, the fact that the antecedent C as well as the consequent D hold simultaneously leads to the natural use of a t-norm. In our case, we use the Gödel t-norm. For example, if an object o_i belongs to a given antecedent in a degree 0.7 and to a given consequent in a degree 0.6, the value that enters to the summation equals to $\min\{0.7, 0.6\} = 0.6$. Summation of such values over all the objects equals to the value a in Table 1.3, the other values from the table are determined analogously.

By using fuzzy sets, we generally get more precise results and we avoid undesirable threshold effects. The further advantage is that the method searches for associations that may be directly interpreted as fuzzy rules for the PbLD inference system, i.e., each association (1.5.1) may be viewed as the fuzzy rule

$$\mathcal{R} := \mathsf{IF}\ (X_1 \text{ is } \mathcal{A}_1)\ \mathsf{AND}\ \cdots\ \mathsf{AND}\ (X_p \text{ is } \mathcal{A}_p)\ \mathsf{THEN}\ (Z \text{ is } \mathcal{B}).$$

The antecedent variables $X \neq X_{i1}, \ldots, X_{ip}$ have the assigned expression `any` – for the

sake of brevity, they are not explicitly expressed in the rule $\mathcal{R}$. All such associations constitute a linguistic description *LD*.

Example 4 *Let us consider Table 1.5.*

	BMI_{ExSm}	...	BMI_{ExBi}	$\text{Chol}_{\text{ExSm}}$	...	$\text{Chol}_{\text{ExBi}}$	BP_{ExSm}	...	BP_{ExBi}
o_1	0		0.9	0.1		0.7	0		0.8
o_2	0		0.2	0.8		0	0.9		0
o_3	0		1	0		1	0		0.9
$\vdots$	$\vdots$		$\vdots$	$\vdots$		$\vdots$	$\vdots$		$\vdots$
o_n	0		0.8	0		0.8	0		0.7

Table 1.5: Example of fuzzy GUHA table.

Depending on the chosen confidence and support degree, the fuzzy GUHA method could generate the following linguistic associations:

$$C(\text{BMI}_{\text{VeBi}}, \text{Chol}_{\text{ExBi}}) \simeq D(\text{BP}_{\text{ExBi}})$$

$$C(\text{BMI}_{\text{SiBi}}, \text{Chol}_{\text{ExBi}}) \simeq D(\text{BP}_{\text{ExBi}})$$

$$C(\text{BMI}_{\text{SiBi}}, \text{Chol}_{\text{SiBi}}) \simeq D(\text{BP}_{\text{VeBi}})$$

$$\ldots$$

$$C(\text{BMI}_{\text{VeBi}}, \text{Chol}_{\text{Bi}}) \simeq D(\text{BP}_{\text{Bi}})$$

$$\ldots$$

that may be viewed and thus, directly represented, as fuzzy rules. For example, the first association would be represented by the following fuzzy rule:

"IF *Body Mass Index is Very Big* AND *Cholesterol is High* THEN *Blood Pressure is High.*"

Chapter 2

Linguistic approach to time series forecasting

In Section 1.1.2, we introduced the potential as well as some drawbacks of computational intelligence methods in time series forecasting.

These observations are among the main motivations for this chapter, which has a fourfold goal:

1. to provide readers with a kind of tasting of distinct methods that may serve as an alternative to standard statistical methods and that may even outperform them;

2. to introduce how these methods may be enhanced, e.g., by using the sensitivity analysis to improve a feature selection for the Support Vector Machine or by a genetic algorithm to search for the optimal Artificial Neural Networks;

3. to introduce purely new combinations of interpretable linguistic fuzzy rules with improved Artificial Neural Networks and Support Vector Machine that provide both – accurate forecasting models and easy to interpret and understand descriptions of the data generating processes;

4. to challenge prior evidence on the inferior forecasting accuracy of Computational Intelligence in operational forecasting [30].

Therefore, we present three Computational Intelligence approaches for multi-step seasonal time series forecasting: the *Automatic Design of Artificial Neural Networks* (ADANN), which uses genetic algorithms to evolve Artificial Neural Networks structures; the *Support Vector Machine with time lag selection based on a sensitivity analysis procedure*; and the *linguistic fuzzy approach* to the trend-cycle analysis and forecasts. The first two methods from different perspectives focus on feature and model selection process for Computational Intelligence methods that is often omitted [31]. These methods will be used to determine the de-trended time series, i.e., to forecast seasonal components of given time series. The latter method focuses on the interpretability issue of fuzzy models describing an forecasting so-called trend-cycle of given time series.

We propose two hybrid combinations of these Computational Intelligence methods, such that the fuzzy approach to trend-cycle forecasts is complemented by the earlier two approaches that forecast seasonal components. The main contribution is the presentation of these methods and the experimental justification of their potential. Besides the achieved high quality accuracy, such models are more easy to interpret by decision-makers when modeling trended series.

2.1 Time series analysis

2.1.1 Time series decomposition

Let

$$\{y_t \mid t = 1, \ldots, T\} \subset \mathbb{R}, \quad T \geq 3 \tag{2.1.1}$$

be a given time series. The task is to analyze it and to forecast its future development, i.e., to determine the values

$$\{y_t \mid t = T+1, \ldots, T+h\} \subset \mathbb{R}, \quad h \geq 1. \tag{2.1.2}$$

There are two standard approaches to this task. The first uses autoregressive and moving averages models from the so-called *Box-Jenkins methodology* [14], and the second approach tries to decompose a given time series into its natural components. The main disadvantage of the Box-Jenkins methodology is that its models (see, for instance, Formula (1.1.2)) are not easily interpretable and, consequently, not as transparent as a decomposition model.

The main idea of the decomposition model (see [13]) is to decompose each element y_t into the following components:

$$y_t = Tr_t + S_t + C_t + E_t \tag{2.1.3}$$

where Tr_t, S_t, C_t, E_t, $t = 1, \ldots, T$ are the *trend, seasonal, cyclic* and *error* components of the time series, respectively.

Remark 2 *Formula (2.1.3) describes so-called additive decomposition. If we use multiplication instead of addition, we obtain multiplicative decomposition. Because the treatment is similar in both cases, for the sake of simplicity, we restrict our focus to the additive case only.*

The name "cyclic" for the C_t component originates in economic cycles, which are not regular. They cyclically replace each other, but they do not maintain constant length, and they depend on many external factors. The error component is a random noise and thus neither practically nor theoretically can be forecasted. In some simplified classical models, these two components are omitted from further consideration.

In the traditional approach, a trend is assumed to be an a priori given function, e.g., linear, polynomial, exponential or a saturation function such as the sigmoidal function. This approach simplifies the analysis, which consists of a regressive determination of parameters of the predetermined function, as well as the forecast, which is a simple prolongation, i.e., an evaluation of the determined trend function at time points $T + 1, \ldots, T + h$.

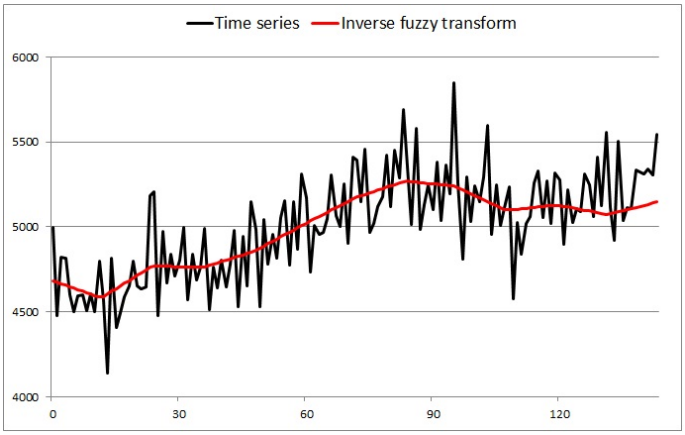


Figure 2.1: Time series with its inverse fuzzy transform. The standard trend analysis would be inappropriate due to irregular cyclic changes.

Such an approach, however, is too restrictive and is thus not always the most appropriate. For example, the trend of the time series in Figure 2.1 can hardly be estimated using a simple function. Because the course of the trend can vary, especially in the case of a long time series, its forecasting is very difficult. Typical examples are equity indexes in which we cannot usually prolong the trend in a simple way because robust growth is often followed by dramatic fall, which can be followed by stagnation and then again by growth. This is directly caused by the influence of the cyclic component. Here, prolongation might, in some cases, be the worst procedure to be applied in forecasting. For such cases, complicated adaptive trend-changing models or models with regime changes [54] are constructed.

Thus, statistical agencies prefer using the so-called *trend-cycle* instead of the trend. According to OECD [1], the trend-cycle is obtained by filtering out the high-frequency fluctuations, i.e., the seasonal and the irregular (noise) components. Following this idea, we propose to use the fuzzy transform method to estimate the trend-cycle because it does not fix any shape for the curve and, moreover, has powerful approximation and noise reduction properties (see [102]). By choosing an appropriate fuzzy partition, we filter out only the high-frequency fluctuations, not the cyclic part of the trend-cycle.

The time series $\{y_t \mid t = 1, \ldots, T\}$ may be viewed as a function y defined on the interval $[0, T]$ that is not given analytically. Instead, measurements $y(t) = y_t$ at points $t = 1, \ldots, T$ are available.

Let us build a uniform fuzzy partition according to Definition 17 such that each of the basic functions $A_2, \ldots, A_{n-1}$ "covers" a number of nodes equal to the number of nodes belonging to a season. For example, in the case of a time series on the monthly basis, each basic function covers 12 points, with the exception of the basic functions A_1, A_n, which cover the first and the last 6 points, respectively. Consequently, the set of points y_t is sufficiently dense with respect to the fuzzy partition. From this point forward, we consider a time series on a monthly basis because everything may be easily generalized for other cases.

Let

$$F[y] = (F_1[y], \ldots, F_n[y]) \tag{2.1.4}$$

be a fuzzy transform of the function y with respect to the given fuzzy partition, and let $\hat{y}$ be its inverse fuzzy transform. The inverse fuzzy transform serves as a model of the trend-cycle. Recall that the shape of the trend-cycle function is not fixed a priori, which would enable us to capture the trend-cycle in a more realistic way.

Remark 3 *Let us note that the fact that the essential role that transparency plays throughout the entire methodology also influences the above choice of constructing basic functions to cover one year of monthly time series. From this point of view, the fuzzy transform values are easily interpretable as average year values. Therefore, a technically possible fuzzy partition with basic functions covering, for instance, 14 values would make no sense. However, a further natural fuzzy partition, for example, with basic functions covering 24 values may make sense and sometimes even improve results.*

The seasonal component S_t from (2.1.3) can be obtained using the formula

$$S_t = y_t - \hat{y}(t), \tag{2.1.5}$$

where $\hat{y}(t) = Tr_t + C_t$. The trend-cycle may be further analyzed and described using autoregressive fuzzy rules; see Section 2.2.

It should be stressed here that the suggested approach is an alternative to the *model with changes in regime* [54] (also called the *regime switching model*). Unlike our approach, that model is based on the theory of random processes and Markov chains. Our motivation is to obtain a transparent description in natural language.

2.2 Trend-cycle forecasting

2.2.1 One step ahead forecasts

The suggested approach to trend analysis also implicitly treats possible cyclic component influences without complicated, adaptive trend-changing mechanisms. This is a significant difference from the classical approach, where we first model only the trend and then determine the seasonal components that are influenced by the irregular cyclic changes. Our approach treats this problem in reverse; the trend-cycle

model $\hat{y}$ primarily serves to obtain pure seasonal components without the cyclic influences.

However, we cannot easily forecast such a trend-cycle model by the prolongation, i.e., by the evaluation of the predetermined fixed trend function at points $t = T + 1, \ldots, T + h$. Due to the drawbacks of this traditional approach, it is not a disadvantage but an advantage, as explained below.

We follow the idea of [101], and for the trend-cycle forecast, we employ perception-based logical deduction (see Subsection 1.3.3). As antecedent variables, we consider the fuzzy transform components of the given time series Y_i, $i = 1, \ldots, n-1$ and their first- and second-order differences:

$$\Delta Y_i = Y_i - Y_{i-1}, \qquad i = 2, \ldots, n-1$$
$$\Delta^2 Y_i = \Delta Y_i - \Delta Y_{i-1}, \qquad i = 3, \ldots, n-1$$

respectively. The fuzzy/linguistic IF-THEN rules take the form

$$\textsf{IF } \Delta^2 Y_{i-1} \text{ is } \mathcal{A}_{\Delta^2 i-1} \textsf{ AND } \Delta Y_{i-1} \text{ is } \mathcal{A}_{\Delta i-1} \textsf{ AND } Y_i \text{ is } \mathcal{A}_i \textsf{ THEN } Y_{i+1} \text{ is } \mathcal{B}_{i+1}. \quad (2.2.1)$$

The differences of the fuzzy transform components expressing the time series trend-cycle of distinct orders are able to describe the dynamics of the time series better than the fuzzy transform components themselves. Furthermore, due to the use of the differences, the time series does not have to be de-trended, as in the case of the classical autoregressive approach using, for example, the ARMA model (1.1.2), nor does it have to be an integrated model, as in the case of the ARIMA.

The rules (2.2.1) may describe logical dependencies of trend-cycle changes (hidden cycle influences), which is highly desirable and suggested in comparison with the standard prolongation of the trend-cycle observed in the past. The advantage of the transparently interpretable form of fuzzy rules using fragments of natural language

is unquestionable. It might be helpful in better understanding the functionalities and motive factors determining the changes in a process yielding the time series in question.

As mentioned before, the fuzzy/linguistic IF-THEN rules of the form (2.2.1) can be understood as a description of an autoregressive process. Every rule describes the local dependence of Y_{i+1} on previous values of Y_i, Y_{i-1}, and so on, expressed in the form of differences ΔY_{i-1}, $\Delta^2 Y_{i-1}$ and the like. The perception-based logical deduction algorithm described in Subsection 1.3.3 then chooses the best local rule or rules with respect to a given situation.

Let us mention that fuzzy rules such as (2.2.1) are automatically generated by the *linguistic learning* algorithm [20] implemented in the software package LFLC 2000 [35] from the fuzzy transform components of the time series and their differences.

Remark 4 *Although the fuzzy/linguistic IF-THEN rules are also generated from the past, the suggested approach can learn, describe, and successfully predict the future of equity indexes mentioned as a motivating example at the beginning of Subsection 2.1.1. Of course, this is possible if similar progress has been observed and measured in the past. The prolongation of a trend function is generally not able to perform this task successfully.*

2.2.2 More steps ahead forecasts with independent models

Let us now consider the fuzzy transform components (2.1.4) of the given time series. Fuzzy/linguistic rules and perception-based logical deduction can be used to forecast the next fuzzy transform components

$$Y_{n+1}, \ldots, Y_{n+\zeta} \tag{2.2.2}$$

from which the trend-cycle of the time series can be determined as values of the inverse fuzzy transform $\hat{y}(T+1), \ldots, \hat{y}(T+h)$, where $\zeta < h$.

It is difficult to forecast the course of the time series for a long time interval. First, note that the fuzzy transform components Y_1, Y_n related to the first and last basic functions are *singular* because they can be depreciated. This property results from the fact that the corresponding basic function of Y_1, like that of Y_n, is only a half of the regular one. Therefore, even the last fuzzy transform component Y_n, which may otherwise be calculated from the given data, is forecasted.

There are two principal ways to forecast the fuzzy transform components:

(i) forecast the next component on the basis of the previous n components (or a subset) and their corresponding first and second differences;

(ii) forecast some of the following components (not necessarily the immediate next one) from some of the components (2.1.4) and their first and second differences.

In case (i), we consider components $Y_1, \ldots, Y_{n-1}$ and their differences $\Delta Y_2, \ldots, \Delta Y_{n-1}$ and $\Delta^2 Y_3, \ldots, \Delta^2 Y_{n-1}$ to forecast the component Y_n. Then, using the same linguistic description, we forecast Y_{n+1} from $Y_1, \ldots, Y_n$, $\Delta Y_2, \ldots, \Delta Y_n$, $\Delta^2 Y_3, \ldots, \Delta^2 Y_n$, and so on.

Obviously, there is a danger of propagating forecast errors because we are forecasting based on forecasted values. The longer the prediction term is, the higher the damage.

Case (ii) overcomes this problem because we build a finite number of independent trend-cycle forecasting linguistic descriptions (models) using the technique described in Subsection 2.2.

The linguistic descriptions consist of rules of the form

$$\mathsf{IF}\ \Delta^2 Y_{i-1}\ \text{is}\ \mathcal{A}_{\Delta^2 i-1}\ \mathsf{AND}\ \Delta Y_{i-1}\ \text{is}\ \mathcal{A}_{\Delta i-1}\ \mathsf{AND}\ Y_i\ \text{is}\ \mathcal{A}_i\ \mathsf{THEN}\ Y_{n+j}\ \text{is}\ \mathcal{B}_{n+j},$$

for suitable $i < n$ and $j = 0, 1, 2, \ldots$. Each linguistic description is generated by the linguistic learning algorithm and each linguistic description may be used to forecast j-steps ahead.

On the basis of the forecasted fuzzy transform components (2.2.2), we can compute the forecasted trend-cycle of the time series, where the latter consists of the values of the inverse fuzzy transform:

$$\hat{y}(T+1), \ldots, \hat{y}(T+h).$$

2.3 Seasonal component forecasting

Since the linguistic fuzzy approach focuses only on modeling and forecasting the trend-cycle of a given time series, the next step in the time series forecasting has to be to forecast the seasonal components separately and to compose them with predicted trend-cycle. The seasonal components may be predicted statistically, as described in [96, 122]. Of course, any other forecasting techniques may be used as well. We propose the use of Computational Intelligence approaches, i.e., the use of Automatic Design of Artificial Neural Networks and Support Vector Machine approaches introduced in the following subsections. The combination of these techniques together with the linguistic approach leads to two novel fuzzy hybrids, termed Fuzzy Artificial Neural Networks (FANN) and Fuzzy Support Vector Machine (FSVM).

2.3.1 Automatic Design of Artificial Neural Networks

Time series processes often exhibit temporal and spatial variability and suffer by issues of nonlinearity of physical processes, conflicting spatial and temporal scale

and uncertainty in parameter estimates. Artificial Neural Networks are flexible models that have the capability to learn the underlying relationships between the inputs and outputs of a process, without needing the explicit knowledge of how these variables are related. We recall typical examples in market predictions [36] or in meteorological [39] and network traffic forecasting [26].

As mentioned above, finding an adequate Artificial Neural Networks model for a particular time series is a key issue. Different studies have treated with the design of an Artificial Neural Networks from three different points of view.

- Connection weights [126, 11]: values for each connection in an Artificial Neural Networks.
- Topology [40, 85]: number of hidden layers, hidden nodes in each layer, etc.
- Learning rules [65]: learning factor and momentum values.

In this thesis, a novel evolving hybrid system that uses both, a genetic algorithm and the backpropagation learning, is proposed. This approach involves an evolution of the Artificial Neural Networks topology and backpropagation learning parameter, with multiple initializations.

Normalization of the time series data has to be done as an initial step and after fitting the Artificial Neural Networks, the inverse process is carried out. This step is important as Artificial Neural Networks with logistic activation functions output values within the range $[0, 1]$. Time series in-samples are transformed into a pattern set with I inputs. A single neuron is placed at the output layer and multi-step forecasts are often performed using an iterative feedback of the previous forecasts [28]. Therefore, each time series is transformed into a patterns set where each

pattern consists of:

$$(N_{t-I}, \ldots, N_{t-2}, N_{t-1}) \rightarrow N_t$$

where all N_i values correspond to the normalized y_i ones. This pattern set is used to train and validate each Artificial Neural Networks generated during the Genetic Algorithm (GA) execution. Thus, the data is split into training (with the first $X\%$ data) and validation sets (with the remaining patterns).

The search for the best Artificial Neural Networks design can be performed by a Genetic Algorithm [38] using exploitation and exploration. When using such Genetic Algorithm, there are three crucial issues:

1. the solution space and what is included into a chromosome;
2. how each solution is codified into a chromosome, i.e. encoding schema;
3. what is the fitness function.

In this work, we opted for a multilayer perceptron as the base forecasting model, with one hidden layer and backpropagation as the learning algorithm, according to [32]. Regarding the backpropagation choice, we note that we use multiple initializations (as distinct seeds are used, see Equation (2.3.1)) and also evolve its learning factor. Under such scheme, backpropagation is unlikely to fall into a local minima. Moreover, backpropagation is the most used algorithm in the time series forecasting domain.

A direct encoding schema for fully connected multilayer perceptron is considered. For this encoding scheme the information placed into the chromosome is: two decimal digits, i.e., two genes to codify the number of inputs nodes (I); two genes for the number of hidden nodes (H); two genes for the learning factor (α); and the last ten genes for the initialization seed (s) value of the connection weights, as the

seed in the Stuttgart Neural Network Simulator (SNNS) [135] is a "long int". This way, the values of I, H, α and s are obtained from the chromosome as follows:

$$\begin{aligned} \textbf{chromosome} &= g_{I_1} g_{I_2} g_{H_1} g_{H_2} g_{\alpha_1} g_{\alpha_2} g_{s_1} g_{s_2} \dots g_{s_{10}} | \forall k, g_k \in \{0, 1, \dots, 9\}, \\ s &= g_{s_1} g_{s_2} \dots g_{s_{10}}, \\ I &= 10 g_{I_1} + g_{I_2} + 1, \\ H &= 10 g_{H_1} + g_{H_2} + 1, \\ \alpha &= (10 g_{\alpha_1} + g_{\alpha_2})/100. \end{aligned} \tag{2.3.1}$$

The search process (Genetic Algorithm) will consist of the following steps:

1. A randomly generated population, i.e., a set of randomly generated chromosomes, is obtained.

2. The phenotypes (Artificial Neural Networks architectures) and fitness value of each individual of the actual generation is obtained. To obtain the phenotype associated to a chromosome and its fitness value:

   (a) The phenotype of an individual of the actual generation is first obtained (using SNNS tool).

   (b) Then for each neural network i, training and validation pattern subsets are obtained from time series data depending on the number of inputs nodes of neural network i.

   (c) The net is trained with backpropagation using SNNS [135]. When the validation error is minimal during the training process, the architecture (topology and weights) is saved – early stopping. This architecture is the final phenotype of the individual.

3. The fitness is the minimum mean square validation error*), during the learning

*)The mean square error in the fitness function is chosen in order to reduce extreme errors that may highly affect multi-step ahead forecasts. Preliminary experiments have shown that this choice leads to the best forecasts.

process.

4. Once the fitness value for whole population is available the Genetic Algorithm operators, namely elitism, selection, crossover and mutation, are applied in order to generate the population of the next generation.

5. Steps 2, 3 and 4 are iteratively executed till a maximal number of generations is reached.

Since the Genetic Algorithm works as a second order optimization procedure, the tuning of its internal parameters is not very crucial, i.e., using a population size of 46, 50 or 54 does not substantially change the results. Based on a few empirical experiments, we set the Genetic Algorithm parameters to: population size, 50; maximum number of generations, 100; percentage of the best individual that stay unchangeable to the next generation (percentage of elitism), 10%; crossover: parents are split in one point randomly selected, offspring are the mixed of each part from parents; mutation probability will be one divided by the length of the chromosome ($1/16 = 0.07$), and it will be carried out for each gene of the chromosome.

2.3.2 Support Vector Machine

The Support Vector Machine is a powerful learning tool based on two key concepts: using a kernel function the Support Vector Machine transforms input variables into a high dimensional feature space and then it finds the best hyperplane to model the data in the feature space.

When applying the Support Vector Machine to time series forecasting, variable (e.g., a time lag) selection process is useful to discard irrelevant time lags in order to obtain a simpler model that is easier to interpret and that usually performs better [28, 56]. Hence, the variable selection process is a critical issue. Additionally, the

Support Vector Machine hyperparameters such as its kernel parameter need to be adjusted [55]. We address this crucial issue by proposing a computationally efficient procedure that performs a simultaneous time lag and the Support Vector Machine model selection for multi-step ahead forecasting.

The Support Vector Machines as any regression algorithm can be applied to time series forecasting by adopting a sliding time window of time lags $\{k_1, k_2, \ldots, k_I\}$, that is used to build a forecast. For a given time period t, the model inputs are $\mathbf{y} = (y_{t-k_I}, \ldots, y_{t-k_2}, y_{t-k_1})$ and the desired output is y_t.

In the Support Vector Machine regression [114], the input ($\mathbf{y}$ with domain Y) is transformed into a high m-dimensional feature space ($\Im$), by using a nonlinear mapping $\phi : Y \rightarrow \Im$ that does not need to be explicitly known but that depends on a kernel function $\kappa(x, x') = \langle \phi(x), \phi(x') \rangle$, where $\langle u, v \rangle$ denotes the inner product of vectors u and v. Then, the Support Vector Machine algorithm finds the best linear separating hyperplane tolerating a small error ε when fitting the data in the feature space:

$$\hat{y}_{t,t-1} = w_0 + \sum_{i=1}^{m} w_i \phi(\mathbf{y}) \tag{2.3.2}$$

where $w_i \in \Re$ are coefficient weights. The ε-insensitive loss function sets an insensitive tube around the residuals and the tiny errors within the tube are discarded.

We adopt the popular gaussian kernel, which presents less parameters than other kernels [125]: $\kappa(x, x') = exp(-\lambda ||x - x'||^2)$, $\lambda > 0$. The Support Vector Machine performance is affected by three parameters: λ, ε and C (a trade-off between fitting the errors and the flatness of the mapping). The kernel parameter λ produces the highest impact in the Support Vector Machine performance, in comparison to C or ε. To reduce the search space, the values are set using the heuristics [22]: $C = 3$ (for a standardized output) and $\varepsilon = \hat{\sigma}/\sqrt{N}$, where $\hat{\sigma} = 1.5/N \times \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$ and

$\hat{y}_i$ is the value predicted by a 3-nearest neighbor algorithm.

Given the setup adopted, the forecasting performance is affected by both time lag and model selection. A better generalization, due to the reduced input space, is achieved if only relevant time lags are fed into the model [56]. Also, if the kernel parameter λ is set with values that are too large or too small, a poor generalization will be achieved.

Sensitivity analysis [70] is a procedure that is applied after the training phase and analyzes the model responses when the inputs change. Let $\hat{y}_{t-k}(j)$ denote the output obtained by holding all input variables at their average values except y_{t-k}, which varies through its entire range with $j \in \{1, \ldots, L\}$ levels. If a given input variable y_{t-k} is relevant then it should produce a high variance V_k. Thus, its relative importance R_k can be given by:

$$\begin{array}{rcl} V_k & = & \sum_{j=1}^{L} (\hat{y}_{t-k}(j) - \overline{\hat{y}_{t-k}}(j))^2/(L-1), \\ R_k & = & V_k / \sum_{i=1}^{I} V_i \times 100\,(\%). \end{array} \tag{2.3.3}$$

This is a simple procedure that only measures single input variance and not interactions of inputs. Yet, even with this limitation, this computationally fast procedure has outperformed other more sophisticated algorithms, e.g., genetic algorithms, for the input variable selection [70].

We propose a simultaneous variable and model selection procedure for multi-step ahead forecasting. The method starts with a maximum of I_{max} time lags and iteratively deletes one input until there are no time lags. The sensitivity analysis is used to select the least relevant lag to be deleted in each iteration, allowing a reduction of the computational effort by a factor of I_{max} when compared to the standard backward selection procedure. Before feeding the Support Vector Machine, all variables are standardized to a zero mean and one standard deviation. After the training, the Support Vector Machine outputs are post-processed with the corresponding inverse

scaling function. During a given iteration, a grid search is used to find the best model hyperparameter $\gamma \in \{2^{-15}, 2^{-13}, \ldots, 2^{1}\}$. The training data is divided into training and validation sets. The former, with 2/3 of the training data, is used to train the Support Vector Machine model. The latter, with the remaining 1/3, is used to select the best model. Similarly to the Automatic Design of Artificial Neural Networks, we adopted the MSE metric for selecting such a model. After the variable and model selection phase, the final model is retrained using all training data (i.e., in-samples). The last known values are fed into the model and multi-step forecasts are built iteratively by using 1-ahead predictions as inputs [26].

The Support Vector Machine experiments were conducted using the **rminer** library[25] of the R tool, which adopts the Sequential Minimal Optimization algorithm to fit the model. In this work, we set $L = 6$ [70] and also, I_{max} was set to $K + 1$ where K denotes the seasonal period (i.e., 13 for monthly time series). The intention is to include all up to the seasonal lag plus an additional one that may be relevant for trended series.

2.4 Results

2.4.1 Time series data sets and evaluation

We deal with seasonal data, since we believe multi-step forecasts are particularly useful for these type of series. Furthermore, seasonal series are commonly present in several domains, such as agriculture, sales, or economy. To compare the proposed forecasting methods, we selected eight benchmark time series (Table 2.4.1).

Seven of them are monthly series from the well-known Hyndman's *Time Series Data Library* [62]:

- **Passengers** data set [14] containing the information about the number (in

Time series	Seasonal period (K)	#in-samples (T)	#out-samples (h_3)
Passengers	12	120	24
Pigs	12	164	24
Cars	12	84	24
Abraham12	12	168	24
Milk	12	144	24
Writing	12	96	24
Cryer7	12	90	24
Mackey-glass	30	731	60

Table 2.1: Time series seasonal period and in-sample/out-sample sizes

thousands) of passengers of international airlines (Jan'49-Dec'60),

- **Pigs** series related to numbers of pigs slaughtered in Victoria (Jan'80-Aug'95),
- **Cars** data consisting of car sales in Quebec ('60-'68),
- **Abraham12** representing gasoline demand at Ontario in millions of gallons ('60-'75),
- **Milk** including monthly milk production in pounds per cow ('62-'75),
- **Writing** containing industry sales for printing and writing paper in thousands of French francs (Jan'63-Dec'72),
- **Cryer7** collecting Portland Oregon average monthly bus ridership divided by one hundred (Jan'73-Jun'82).

All these seven data sets contain real-world data from different areas, which makes them interesting to forecast. First, because accurate forecasts can have an impact in their application domains. Second, these data sets suffered indirectly from external and dynamic phenomena, such as weather, economic or technological conditions that are more difficult to predict.

The last series, called **Mackey-glass**, is based on the Mackey-Glass differential equation [46] and it is widely regarded as a benchmark for comparing the generalization ability of different methods. This series is a chaotic time series generated from a time-delay ordinary differential equation. This time series has been chosen in order to extend the experimental data sets by a different kind of a benchmark, i.e., by a time series that is not based on real-world data, that is not on a monthly basis, and that contains neither a trend nor a noise component.

To evaluate of the global performance of a forecasting model, we apply SMAPE and MASE forecasting errors (see Subsection 1.1.3). It is worth recalling that forecasting accuracy depends upon forecasting horizons [30]. Thus, we opted to compute errors for three distinct forecasting horizons: $h_1 = K$; $h_2 = 1.5K$; and $h_3 = 2K$, where K is the seasonal period. The K, T and h_3 values for the eight benchmark series are presented in Table 2.4.1.

2.4.2 ARIMA by ForecastPro® as a comparison benchmark

In order to present the results with a clear insight of how good they are, a well-known method is used as a comparison benchmark. We chose the seasonal variant of the very popular seasonal Autoregressive Integrated Moving Average (ARIMA) model [14]. Note, that in order to avoid any bias from a naive implementation of ARIMA, we adopted the ForecastPro® (FP) [47] professional forecasting software. In particular, the tool was fed with the in-samples of the data sets from Table 2.4.1 and executed the full automatic parameter selection of ARIMA to obtain the forecasts. This automatic selection includes the search for the best ARIMA variant, including its internal parameters, and detection of events such as level shifts or outliers.

The choice of FP ARIMA was straightforward because of several reasons. First, all presented Computational Intelligence methods are also of autoregressive nature.

Second, the chosen benchmark is by far better than standard ARIMA. This is mainly due to the implementation by ForecastPro® enhanced by above mentioned events detection and optimization which make the FP ARIMA a method that is difficult to outperform. This is underlined by the fact that other methods, such as exponential smoothing or mathematical curve fitting, were also tested as possible benchmarks but did not outperform FP ARIMA. Third, the automatic FP ARIMA is a popular tool that is at disposal of many forecasting professionals, who may easily check the results. Moreover, comparison to a such widely used tool has a significant explanatory value, which is fully coherent with principles of evaluating method [5]. Finally, latest advanced methods have no standardized implementations and thus, one risks that the results highly depends rather on the particular chosen implementation than on the potential of the method itself.

2.4.3 Forecasting performance

First, we analyze the performance of the fuzzy hybrids Fuzzy Artificial Neural Networks and Fuzzy Support Vector Machine from Section 2.3, when compared with the automatic ARIMA (ForecastPro). Accuracy was measured on all three forecasting horizons h_1, h_2, h_3 by both SMAPE and MASE (Table 2.4.3 and 2.4.3).

For both SMAPE and MASE metrics, Fuzzy Artificial Neural Networks obtains the best results for Passengers, Abraham12, Mackey-glass and partly also for Writing (for h_2 and h_3). ARIMA performed generally best for Pigs, Cars, Cryer7 and partly also for Milk (for h_1 and h_3). Fuzzy Support Vector Machine wins only in a single case.

The overall comparison is performed using the arithmetic mean and median (over all series, last rows of Tables 2.4.3 and 2.4.3) for both metrics and all horizons. When compared with the arithmetic mean, the median is more robust with respect

Series	Horizon								
	h_1			h_2			h_3		
	FP	FANN	FSVM	FP	FANN	FSVM	FP	FANN	FSVM
Passengers	6.5	**2.1**	3.0	7.3	**2.6**	3.8	8.0	**2.5**	3.9
Pigs	**6.1**	6.7	7.7	**6.1**	6.7	7.9	**7.1**	8.2	7.8
Cars	**7.4**	12.1	11.6	**8.4**	10.5	10.3	**9.1**	10.0	10.9
Abraham12	5.5	**4.0**	4.8	6.2	**5.1**	5.5	6.2	**5.6**	5.9
Milk	**0.8**	1.1	1.0	1.0	1.1	**0.9**	**0.9**	1.1	1.0
Writing	**7.3**	8.8	7.5	9.0	**8.0**	9.0	9.9	**8.7**	9.9
Cryer7	**9.0**	16.1	14.1	**12.1**	18.4	17.5	**13.8**	18.5	17.7
Mackey-glass	22.7	**3.9**	6.8	21.5	**3.9**	10.5	26.2	**9.6**	19.0
Mean	8.2	**6.9**	7.1	9.0	**7.0**	8.2	10.2	**8.0**	9.5
Median	6.9	**5.4**	7.2	7.9	**5.9**	8.5	8.6	**8.5**	8.9

Table 2.2: Comparison of Fuzzy Artificial Neural Networks (FANN), Fuzzy Support Vector Machine (FSVM) and ForecastPro (FP) (SMAPE, best values in **bold**)

to outliers. Fuzzy Artificial Neural Networks is ranked at first place with respect to SMAPE for all horizons, followed by Fuzzy Support Vector Machine. A similar observation is found for median values of MASE errors with the only change that Fuzzy Support Vector Machine shares the best median with Fuzzy Artificial Neural Networks for h_1. Thus we can state that although ARIMA performed best for half of the series considered, its accuracy for the remaining series was not that stable, when compared with the other two methods, yielding an overall mean and median that globally ranks this method at third place. However, taking into account only the arithmetic mean of errors measured by MASE, then ARIMA outperforms both fuzzy hybrids although not significantly. More detailed discussion will be provided in Section 2.4.5.

2.4.4 Interpretability of fuzzy rules

Interpretability is often assumed to be a key feature (and advantage) of fuzzy models in various areas of application [21]. However, this aspect of fuzzy models is sometimes overused. Undoubtedly, there is a significant difference between rather

Series	Horizon								
	h_1			h_2			h_3		
	FP	FANN	FSVM	FP	FANN	FSVM	FP	FANN	FSVM
Passengers	1.24	**0.39**	0.56	1.40	**0.50**	0.75	1.60	**0.49**	0.80
Pigs	**0.64**	0.70	0.80	**0.64**	0.71	0.81	**0.76**	0.87	0.81
Cars	**0.54**	0.79	0.75	**0.62**	0.71	0.69	**0.66**	0.75	0.74
Abraham12	1.12	**0.80**	0.96	1.23	**1.01**	1.09	1.27	**1.15**	1.22
Milk	**0.19**	0.24	0.22	0.22	0.25	**0.21**	**0.21**	0.24	0.22
Writing	**0.47**	0.61	0.52	0.63	**0.58**	0.65	0.69	**0.62**	0.69
Cryer7	**3.06**	5.66	4.91	**4.11**	6.41	6.09	**4.77**	6.52	6.18
Mackey-glass	1.30	**0.22**	0.36	1.29	**0.23**	0.58	1.48	**0.49**	1.03
Mean	**1.07**	1.18	1.14	**1.27**	1.30	1.36	1.43	**1.39**	1.46
Median	0.88	**0.66**	**0.66**	0.94	**0.65**	0.72	1.02	**0.69**	0.81

Table 2.3: Comparison of Fuzzy Artificial Neural Networks (FANN), Fuzzy Support Vector Machine (FSVM) and ForecastPro (FP) (MASE, best values in **bold**)

numerically oriented fuzzy models such as the Takagi-Sugeno rules and models that are, say, more linguistically oriented, such as fuzzy rules with fuzzy sets that interpret both antecedents and consequents. But even in the latter case there are fundamental differences. For example, a misleading interpretation of conjunctive (Mamdani-Assilian) rules as fuzzy IF-THEN rules, although their meaning is rather different [34, 91], is a common weakness. In addition, even if the interpretation is correct, some types of treatment of the interpretations of linguistic labels with several parameters may lead to something that is very far from anything that may be called "linguistic".

The previous sentence aims at well-tuned fuzzy models constructed with help of various tuning strategies leading to black-box functions that disregard the importance of interpretability. Let us recall the following crucial idea [12]: "*one may argue that proper input-output behavior is the central goal of automatic tuning. To some extent, this is true; however, this is not the primary mission of fuzzy systems.*" This idea perfectly addresses the time series forecasting. Even here, the accuracy of forecasts is undoubtedly the key issue. Nevertheless, we have to keep in mind

the motivation behind using a fuzzy model, which generally assumed to provide an interpretable, transparent and understandable model rather than to follow only optimality goals.

We do not claim that fuzzy models should not be precise. On the contrary, fuzzy models seem to be very promising within the forecasting area so far and any forecasting model, including a fuzzy one, should perform the time series forecasting task with high accuracy. The goal is an interpretable model that does not necessarily "*lead to a painful loss of accuracy*" [12].

The key issue in maintaining the interpretability even in the case of a tuned fuzzy model, should be the fulfillment of several constrains on fuzzy sets that interpret linguistic expressions. Namely, they should be ordered according to natural order of linguistic expressions. That is, the interpretation of *small* should placed to the left of the interpretation of *medium* and so on. In addition, they should be convex and form a partition of the universe. Let us stress, that these constraints are fully consistent with the theory of evaluative expressions based on the basic trichotomy of *small*, *medium* and *big* and the ordering of linguistic hedges.

A similar idea is adopted in [103] where authors claim that their tuning method does not modify the initial partition in a severe manner (and interpretability is thus kept), because the widths of membership functions change by 12.9% on average and their centers change by 3.1%. Membership functions of fuzzy sets assigned to linguistic expressions in the approach discussed in this chapter do not change at all. Thus, an interpretation of each linguistic expression is the same anywhere in any linguistic description.

To underline the interpretability and the linguistic nature of evaluative expressions and the used fuzzy/linguistic IF-THEN rules, we present one of the generated models. Let us consider the **Pigs** time series. In addition to the forecast itself, a

Nr.	Antecedents				Consequent
	Y_i	ΔY_i	ΔY_{i-1}	$\Rightarrow$	ΔY_{i+1}
1	Bi	QR Sm	Ex Sm	$\Rightarrow$	QR Sm
2	QR Bi	−Ro Bi	QR Sm	$\Rightarrow$	−Si Bi
3	Ex Bi	QR Sm	QR Sm	$\Rightarrow$	−Ro Bi
4	Ro Bi	Ex Sm	Sm	$\Rightarrow$	QR Sm
5	Ze	−Ex Bi	−Ro Bi	$\Rightarrow$	Ex Sm
6	Ex Sm	Ex Sm	−Ex Bi	$\Rightarrow$	Ve Sm
7	Si Sm	Ve Sm	Ex Sm	$\Rightarrow$	Sm
8	Sm	Sm	Ve Sm	$\Rightarrow$	QR Sm
9	QR Sm	QR Sm	Sm	$\Rightarrow$	QR Bi
10	QR Bi	QR Bi	QR Sm	$\Rightarrow$	−Ex Sm

Table 2.4: Fuzzy rules generated for the description and prediction of *Pigs* time series. Abbreviations of evaluative expressions can be found in Section 1.3.

user is provided by the linguistic description composed of ten fuzzy rules symbolically displayed in Table 2.4.4. As we can see, all of the rules are purely linguistic – all the antecedents and consequents are linguistic evaluative expressions according to the respective theory.

Thus, every single fuzzy rule can indeed be taken as a sentence in natural language. For instance, consider the very first fuzzy rule:

IF Y_i is Bi AND ΔY_i is QR Sm AND ΔY_{i-1} is Ex Sm THEN ΔY_{i+1} is QR Sm.

It may be read as follows:

If the number of pigs slaughtered in the current year is big and the biannual increment is quite roughly small and the previous biannual increment was also positive with extremely small strength then the upcoming biannual increment will be quite roughly small.

Hence, such a rule may be understood as follows. Given a big number of slaughtered pigs and with increasing and slight increasing trend from the last observation the

increase will not finish but will continue with quite roughly slight strength.

Similarly, we can consider the second fuzzy rule where one can find an information that having quite roughly big number of slaughtered pigs with trend that changed its direction from (quite roughly) small increment to (roughly) big decrease signalizes that the trend numbers really reached a kind of saturation of the market and the number of slaughtered pigs will continue in a strong decrease.

We claim, that such readable information is an additional value that might be very helpful (e.g., to check if the model makes sense within the domain) for further decision-making and management processes. This is particularly useful for critical domain applications (e.g., control or medicine).

2.4.5 Discussion

When analyzing the obtained results in Subsection 2.4.3, it is clear that the combined methods Fuzzy Artificial Neural Networks and Fuzzy Support Vector Machine in overall evaluations (means and medians with respect to both metrics) outperform the benchmark with the exception – means of error measured by MASE. Observing Table 2.4.3, it is clear that the reason lies in the inaccurate predictions of Fuzzy Artificial Neural Networks and Fuzzy Support Vector Machine in one series – Cryer7. And this time series has significantly higher influence on the overall evaluation measured by MASE than the other series. And as stated above, the arithmetic mean is more sensitive to such outliers when compared with the median. It is also interesting to note that measured by SMAPE, Cryer7 is not that much significant in the overall evaluation. This confirms the necessity of using more that just one accuracy metric that can lead to misleading conclusions.

Since the Automatic Design of Artificial Neural Networks as well as the Support Vector Machine performed well for Cryer7, it is the fuzzy approach forecasting the

trend-cycle that is responsible for the weak forecasting performance of Fuzzy Artificial Neural Networks and Fuzzy Support Vector Machine. This fact can be visually observed from Figure 2.2. The problem is that there is a change in the trend-cycle development that has not been observed before and thus, can hardly be predicted. The top element of the so far nearly constantly increasing Cryer7 series is only three values before the end of in-samples set. The last three decreasing values are sufficient for the Automatic Design of Artificial Neural Networks and the Support Vector Machine methods which underlines their flexibility but rather insufficient for the fuzzy approach and enhanced FP ARIMA that forecast a continually increasing trend.

In the case of the fuzzy approach, the problem is that it takes into account the components of the F-transform and these are average values. Last three decreasing values do not change the whole component sufficiently in order to provide an evidence of a decreasing trend-cycle. This is a common weakness of any method using aggregated values (recall e.g. PAA – the Piecewise Aggregate Approximation [69]) in case of an unlucky placement of the border point between the in-sample set and the out-sample set.

Moreover, if we artificially delete the Cryer7 time series, Fuzzy Artificial Neural Networks as well as Fuzzy Support Vector Machine outperform not only the FP ARIMA but for some horizons also their related individual methods Automatic Design of Artificial Neural Networks and Support Vector Machine. So, it is a harmony of several conditions (unobserved change in the trend-cycle development; specific placement of the border between in-samples and out-samples; too significant influence of one series to overall results with respect to a single accuracy metric) that leads to the overall evaluation that does not favor the fuzzy hybrids in comparison to the benchmark when using mean and MASE.

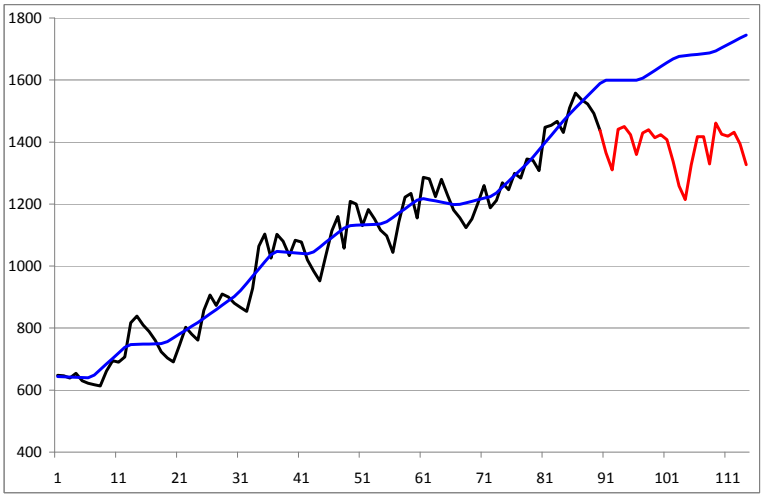


Figure 2.2: Graph of Cryer7 time series. Black line depicts the in-samples, red line depicts the out-samples, blue line depicts the trend-cycle including its prediction.

On the other hand, we have to stress that the proposed fuzzy approach is targeted for trended series. For comparison purposes, we applied the fuzzy variants (Fuzzy Artificial Neural Networks and Fuzzy Support Vector Machine) to the stationary Mackey-glass series, although it makes no sense to describe linguistically a trend for such series.

Chapter 3

Redundancies in systems of fuzzy/linguistic IF-THEN rules

3.1 Basic concepts

Linguistic descriptions may suffer from several problems, such as redundancy or inconsistency. The danger of an existence of these problems is even strengthened in cases when the fuzzy rule base is automatically generated from data. For example, the redundancy, i.e., the existence of redundant fuzzy rules in a given linguistic description, is usually caused by redundant measurements that are used to generate a linguistic description, see the properties of the learning procedure in [20]. Such situations are not typical only for fuzzy control but also for further applications, such as analysis and forecast of time series with the help of fuzzy rules, see [96, 122].

Generally, redundancy in fuzzy rules is observed as an existence of fuzzy rules with distinct but not contradictory antecedents (an antecedent is fully overlapped by an antecedent of another rule) and identical consequents. However, we will show that the situation is not straightforward and that the redundancy phenomenon requires further formal investigation.

Remark 5 *Let us fix the notation for the rest of this thesis. Let us consider a linguistic description LD and let us be given an observation* u_0 *in a given context* $w \in W$. *Let* C *be the conclusion derived from* u_0 *based on LD using the rule of perception based logical deduction given by (1.3.13). Then this fact will be denoted by*

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : C. \tag{3.1.1}$$

Note that C *is a set of fuzzy sets, in general. By writing, e.g.,* $C = D$ *we are expressing the fact that sets* C *and* D *are equal, i.e., they have precisely the same elements (fuzzy sets).*

In the following, we will suppose that linguistic descriptions under consideration contain at least two fuzzy/linguistic IF-THEN rules, i.e., $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ and $m \geq 2$.

Definition 21 Let $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ be a linguistic description (1.3.7). The rule $\mathcal{R}_i$ is *redundant* in LD if $D_1 = D_2$ for each value $u_0 \in w$, $w \in W$, where

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : D_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0, w)\right) : D_2$$

and $LD' = LD \setminus \{\mathcal{R}_i\}$.

As we have mentioned, redundancy is observed as an existence of fuzzy rules with distinct overlapping antecedents and identical consequents. However, as we will show below, sometimes such an intuitively redundant fuzzy rule does not have to be always redundant with respect to a formal definition of the redundancy. Therefore, such a rule will be called *suspected of redundancy* and a further analysis of its potential redundancy is necessary.

Definition 22 Let *LD* be a linguistic description (1.3.7), let $\{\mathcal{R}_i, \mathcal{R}_j\} \subseteq LD$, $i \neq j$. The rule $\mathcal{R}_i$ is *suspected of redundancy with respect to* $\mathcal{R}_j$ (denoted by $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$) if $C_1 = C_2$ for each value $u_0 \in w$, $w \in W$, where

$$r_{PbLD}\left(LPerc^{\{\mathcal{R}_i,\mathcal{R}_j\}}(u_0, w)\right) : C_1$$

and

$$r_{PbLD} : \left(LPerc^{\{\mathcal{R}_j\}}(u_0, w)\right) : C_2.$$

Theorem 3.1.1

Let LD be a linguistic description (1.3.7), let $\{\mathcal{R}_i, \mathcal{R}_j\} \subseteq LD$. *The rule* $\mathcal{R}_i$ *is suspected of redundancy with respect to* $\mathcal{R}_j$ *if and only if* $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$ *and* $\mathcal{B}_i = \mathcal{B}_j$.

PROOF: Let $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$. Then there exists no $w \in W$ and $u_0 \in w$ such that

$$LPerc^{\{\mathcal{R}_i,\mathcal{R}_j\}}(u_0, w) = \{A_i, A_j\}.$$

Let us discuss distinct situations with respect to $u_0 \in w$.

(i) If $u_0 \in w$ is such that $A_i(u_0) = A_j(u_0) = 0$ then $C_1 = \emptyset$ as well as $C_2 = \emptyset$ and hence $C_1 = C_2$.

(ii) If $u_0 \in w$ is such that $A_i(u_0) < A_j(u_0)$ then $LPerc^{\{\mathcal{R}_i,\mathcal{R}_j\}}(u_0, w) = \{A_j\}$ and therefore

$$C_1 = \{A_j(u_0) \to B_j(v) \mid v \in w'\}, \quad C_2 = \{A_j(u_0) \to B_j(v) \mid v \in w'\}$$

and thus, $C_1 = C_2$.

(iii) If $u_0 \in w$ is such that $A_i(u_0) = A_j(u_0)$ then necessarily $A_i(u_0) = A_j(u_0) = 1$ and together with $\mathcal{A}_i <_{LE} \mathcal{A}_j$ it yields $LPerc^{\{\mathcal{R}_i,\mathcal{R}_j\}}(u_0, w) = \{A_i\}$. Therefore, we may continue with

$$C_1(v) = A_i(u_0) \to B_i(v) = 1 \to B_i(v) = B_i(v)$$

and with

$$C_2(v) = A_j(u_0) \to B_j(v) = 1 \to B_j(v) = B_j(v),$$

which by the assumption $B_i = B_j$ gives $C_1 = C_2$.

Let us prove the opposite direction, i.e., that assuming $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ necessarily leads to $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$ and $B_i = B_j$. Let us assume $\mathcal{A}_i \not\leq_{LE} \mathcal{A}_j$ or $B_i \neq B_j$. If $\mathcal{A}_i \not\leq_{LE} \mathcal{A}_j$ then for each $w \in W$ there exists $u_0 \in w$ such that $A_j(u_0) < A_i(u_0)$ which yields $LPerc^{\{\mathcal{R}_i,\mathcal{R}_j\}}(u_0, w) = \{A_i\}$. Therefore, we may continue with

$$C_1(v) = A_i(u_0) \to B_i(v), \quad C_2(v) = A_j(u_0) \to B_j(v)$$

and thus $C_1 \neq C_2$ which is in a contradiction with the assumption $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$.

Thus only the case $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$ remains. Even if it holds but $B_i \neq B_j$ then for each $w \in W$ there exists $u_0 \in w$ such that

$$A_j(u_0) = A_i(u_0) = 1$$

and for such u_0

$$C_1 = \{B_i\}, \quad C_2 = \{B_j\}$$

which together with $B_i \neq B_j$ leads to $C_1 \neq C_2$ which is in a contradiction with the assumption $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. □

Theorem 3.1.1 claims that a fuzzy rule with an antecedent overlapped by an antecedent of another rule with the identical consequent is suspected of redundancy with respect to that rule. Furthermore, there are no other fuzzy rules that could be suspected of redundancy with respect to another fuzzy rule besides those that meet the above mentioned situation. In other words, Theorem 3.1.1 specifies fuzzy rules that makes sense to investigate. The situation is displayed on Figure 3.1 where one can see that nothing changes if fuzzy rule $\mathcal{R}_i$ is removed, unless other rules are involved.

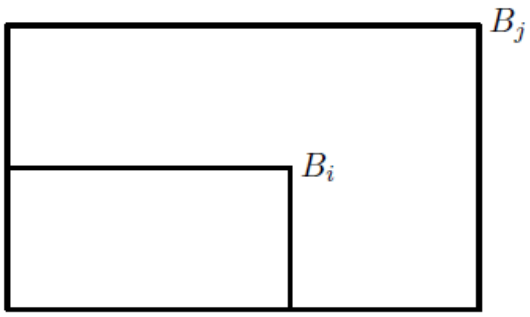


Figure 3.1: Visualization of the fuzzy rule $\mathcal{R}_i$ that is suspected of redundancy with respect to the fuzzy rule $\mathcal{R}_j$. Displayed rectangles symbolically delimit areas where the respective fuzzy rules fire. Both rectangles are black and solid to symbolize that $\mathcal{B}_i = \mathcal{B}_j$.

Lemma 1 *The binary operation $\hookrightarrow$ is a strict partial order.*

PROOF: It follows directly from Definition 22 that $\hookrightarrow$ is irreflexive. Due to Theorem 3.1.1, the proof of the lemma then reduces only to the proof of the fact that $\leq_{\mathrm{LE}}$ is antisymmetric and transitive. All these properties can be easily shown. □

3.2 Detection of suspected rules and their possible cancellation

In Theorem 3.1.1, we have specified the rules suspected of redundancy. However, due to the involvement of other rules, the suspected rules do not have to be necessarily redundant which may be demonstrated easily. Let us consider a linguistic description LD with $\{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\} \subseteq LD$ where $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$, the antecedents are ordered as follows $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_j$ and the consequent $\mathcal{B}_k$ is different from the consequents $\mathcal{B}_i = \mathcal{B}_j$. Then, the fuzzy rule $\mathcal{R}_k$ "cancels" the suspected redundancy of $\mathcal{R}_i$. For a

visualization of such a situation we refer to Figure 3.2.

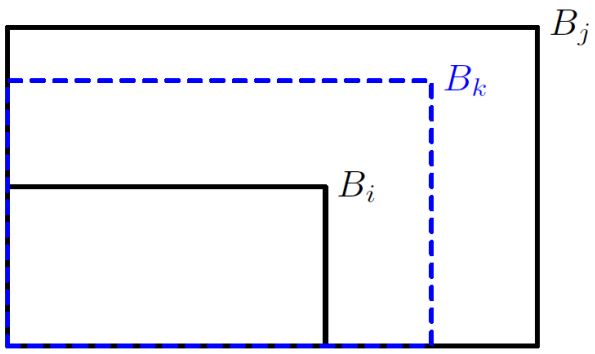


Figure 3.2: Visualization of a situation when the fuzzy rule $\mathcal{R}_k$ "cancels" the potential redundancy of the fuzzy rule $\mathcal{R}_i$ with respect to $\mathcal{R}_j$. The area where $\mathcal{R}_k$ fires is delimited by blue dashed line to show that $\mathcal{B}_k \neq \mathcal{B}_i(\mathcal{B}_j)$.

Naturally, one could release a hypothesis describing the situation in which fuzzy rule $\mathcal{R}_i$ which is suspected of redundancy with respect to $\mathcal{R}_j$ is not really redundant. Formally, this hypothesis could be formulated as follows.

Hypothesis 1 *Let $\{\mathcal{R}_i, \mathcal{R}_j\} \subseteq LD$ and let $\mathcal{R}_i$ be suspected of redundancy with respect to $\mathcal{R}_j$. If there exists a rule $\mathcal{R}_k \in LD$ such that*

(1) $\mathcal{B}_k \neq \mathcal{B}_i$,

(2) $\mathcal{A}_k \leq_{\text{LE}} \mathcal{A}_j$,

and either

(3a) $\mathcal{A}_i \leq_{\text{LE}} \mathcal{A}_k$,

or

(3b) $\mathcal{A}_i \parallel_{\text{LE}} \mathcal{A}_k$, *(where $\parallel_{\text{LE}}$ denotes the incomparability)*

then $\mathcal{R}_i$ is NOT redundant in LD.

However, after a careful investigation, one may find a counterexample that refutes Hypothesis 1. It is sufficient to consider another fuzzy rule $\mathcal{R}_p \in LD$ with the consequent $\mathcal{B}_p = \mathcal{B}_i$ and with an appropriate antecedent, see Figure 3.3. In this case, fuzzy rule $\mathcal{R}_k \in LD$ cancels the suspicion of the fuzzy rule $\mathcal{R}_i$, but the fuzzy rule $\mathcal{R}_p \in LD$ cancels the cancellation provided by $\mathcal{R}_k \in LD$. Thus, Hypothesis 1 does not hold.

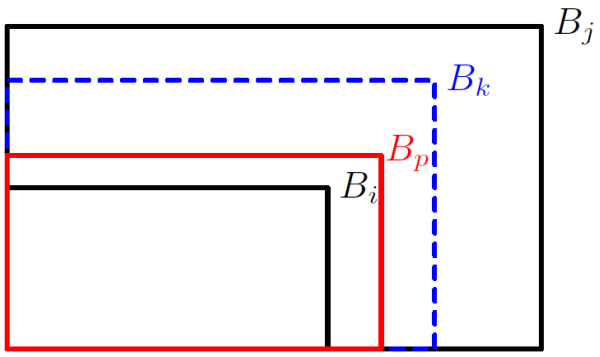


Figure 3.3: Scheme showing fuzzy rules that contradict Hypothesis 1 because $\mathcal{R}_p$ with $\mathcal{B}_P = \mathcal{B}_i(\mathcal{B}_j)$ cancels the cancellation of $\mathcal{R}_k$.

Nevertheless, Hypothesis 1 may be rewritten into a valid theorem if we consider a linguistic description with three rules only.

Theorem 3.2.1

Let $LD = \{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\}$ and let $\mathcal{R}_i$ be suspected of redundancy with respect to $\mathcal{R}_j$. If either (1), (2), (3a) or (1), (2), (3b) from Hypothesis 1 hold, then $\mathcal{R}_i$ is NOT redundant in LD.

PROOF: Let us denote

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : C_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0, w)\right) : C_2$$

where $LD' = LD \setminus \{\mathcal{R}_i\}$.

Let *(1)-(3a)* or *(1)-(3b)* hold. Due to the fact that $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ and due to Theorem 3.1.1 we know that $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_j$. Furthermore, due to $\mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_j$ (from *(2)*) we know that $\mathcal{A}_i$ and $\mathcal{A}_k$ are of the same type (they have identical atomic evaluative expressions) and thus, for any $w \in W$ there exists $u_0 \in w$ such that

$$A_i(u_0) = A_k(u_0) = 1.$$

Further, from *(3a)* or *(3b)* we know that either $\mathcal{A}_k \parallel_{\mathrm{LE}} \mathcal{A}_i$ or $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_k$. In the first case

$$LPerc^{LD}(u_0, w) = \{A_i, A_k\}, \quad \text{but}$$
$$LPerc^{LD'}(u_0, w) = \{A_k\} \quad (\text{because } \mathcal{R}_i \notin LD')$$

Therefore $C_1 = \{B_i, B_k\}$, but $C_2 = \{B_k\}$ and because of $\mathcal{B}_k \neq \mathcal{B}_i$ (from *(1)*), we immediately get $C_1 \neq C_2$.

In the second case

$$LPerc^{LD}(u_0, w) = \{A_i\}, \quad \text{but}$$
$$LPerc^{LD'}(u_0, w) = \{A_k\} \quad (\text{because } \mathcal{R}_i \notin LD')$$

Therefore $C_1 = \{B_i\}$, but $C_2 = \{B_k\}$ and because of $\mathcal{B}_k \neq \mathcal{B}_i$ (from *(1)*), we immediately get $C_1 \neq C_2$ which completes the proof. □

There exists another hypothesis which seems to be naturally valid. Let a linguistic description LD be given , let $\{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\} \in LD$ and let $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. If $\mathcal{A}_k \parallel_{\mathrm{LE}} \mathcal{A}_j$

but they are of the same type (they use the same atomic expression) and $\mathcal{B}_k$ is different from $\mathcal{B}_i = \mathcal{B}_j$ then rule $\mathcal{R}_k$ "cancels" the redundancy that $\mathcal{R}_i$ was suspected to possess, see Figure 3.4.

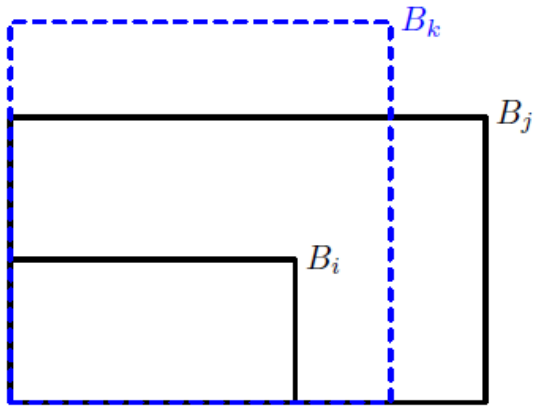


Figure 3.4: Visualization of another situation when the fuzzy rule $\mathcal{R}_k$ "cancels" the potential redundancy of the fuzzy rule $\mathcal{R}_i$ with respect to $\mathcal{R}_j$. The area where $\mathcal{R}_k$ fires is delimited by blue dashed line to show that $\mathcal{B}_k \neq \mathcal{B}_i(\mathcal{B}_j)$.

Hypothesis 2 *Let $\{\mathcal{R}_i, \mathcal{R}_j\} \subseteq LD$ and let $\mathcal{R}_i$ be suspected of redundancy with respect to $\mathcal{R}_j$. If there exists a rule $\mathcal{R}_k \in LD$ such that*

(4) $\mathcal{B}_k \neq \mathcal{B}_i$,

(5) $\mathcal{A}_k \|_{\mathrm{LE}} \mathcal{A}_j$, *but* $\mathcal{A}_k, \mathcal{A}_j$ *have the same atomic expression,*

(6) $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_k$,

then $\mathcal{R}_i$ is NOT redundant in LD.

In both Hypothesis 1 and Hypothesis 2, it is necessary to assume the consequents of cancelling rules $\mathcal{B}_k$ to be different than the consequent $\mathcal{B}_i$ that appears in fuzzy rules $\mathcal{R}_i$ as well as $\mathcal{R}_j$. This assumption is specified by conditions *(1)* and *(4)*,

respectively. The difference between Hypothesis 1 and Hypothesis 2 consists in the "placement" of the cancelling fuzzy rule $\mathcal{R}_k$ in a sense of determining the area (with respect to $\mathcal{R}_i$ and $\mathcal{R}_j$) where this rule fires. In the case of Hypothesis 1, the cancelling rule $\mathcal{R}_k$ fires in an area that is fully embedded into the area where $\mathcal{R}_j$ fires, which is given by condition *(2)*. More specifically, the antecedent of $\mathcal{R}_j$ fully overlaps the antecedent of $\mathcal{R}_k$, see Figure 3.2. Moreover, the antecedent of $\mathcal{R}_k$ must not be fully overlapped by the antecedent of $\mathcal{R}_i$.

In case of Hypothesis 2, the situation is different. The antecedent of $\mathcal{R}_k$ neither overlaps the one of $\mathcal{R}_j$ nor is overlapped by it, see Figure 3.4. This is given by condition *(5)*. On the other hand, the antecedent $\mathcal{R}_k$ must necessarily overlap the antecedent of $\mathcal{R}_i$, which is given by condition *(6)*. This is a significant difference to condition *(3b)* from Hypothesis 1 which allowed also incomparable position instead of the full overlap.

Analogously to the case of Hypothesis 1, neither Hypothesis 2 holds, see the counterexample on Figure 3.5. There can exist a rule $\mathcal{R}_p \in LD$ with the consequent $\mathcal{B}_p = \mathcal{B}_i$ and with an appropriate antecedent that cancels the cancellation of the suspicion of the rule $\mathcal{R}_i$ provided by rule $\mathcal{R}_k$.

However, Hypothesis 2 may be rewritten into a valid theorem where we again consider a linguistic description that consists of just three rules.

Theorem 3.2.2

Let $LD = \{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\}$ and let $\mathcal{R}_i$ be suspected of redundancy with respect to $\mathcal{R}_j$. If for $\mathcal{R}_k$ conditions (4), (5) and (6) from Hypothesis 2 hold, then $\mathcal{R}_i$ is NOT redundant in LD.

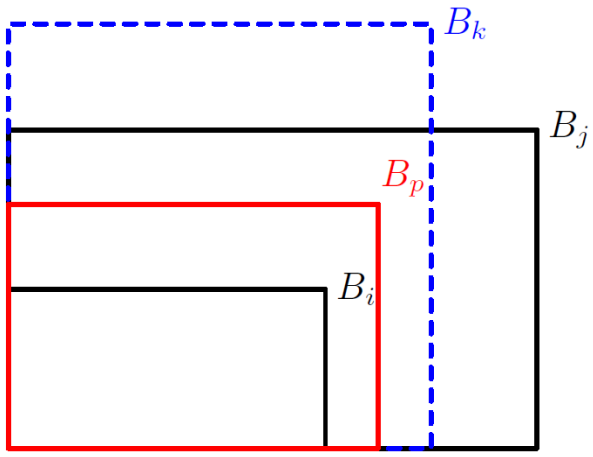


Figure 3.5: Scheme showing fuzzy rules that contradicts Hypothesis 2 because $\mathcal{R}_p$ with $\mathcal{B}_P = \mathcal{B}_i(\mathcal{B}_j)$ cancels the cancellation of $\mathcal{R}_k$.

PROOF: Let us denote

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : C_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0, w)\right) : C_2$$

where $LD' = LD \setminus \{\mathcal{R}_i\}$.

Let *(4)-(6)* hold. Due to the fact that $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ and due to Theorem 3.1.1 we know that $\mathcal{A}_i \leq_{\text{LE}} \mathcal{A}_j$. Furthermore, due to $\mathcal{A}_i \leq_{\text{LE}} \mathcal{A}_k$ (from *(6)*) we know that for any $w \in W$ there exists $u_0 \in w$ such that

$$A_i(u_0) = A_k(u_0) = A_j(u_0) = 1.$$

Furthermore, $LPerc^{LD}(u_0, w) = \{A_i\}$ but since $\mathcal{R}_i \notin LD'$ and due to $\mathcal{A}_k \parallel_{\text{LE}} \mathcal{A}_j$ (from *(5)*), $LPerc^{LD'}(u_0, w) = \{A_k, A_j\}$.

Therefore $C_1 = \{B_i\}$, but $C_2 = \{B_k, B_j\}$ and because of $\mathcal{B}_k \neq \mathcal{B}_i$ (from *(4)*), we obtain $C_1 \neq C_2$ which completes the proof. □

Theorem 3.2.1 and Theorem 3.2.2 were formulated for a linguistic description that consist of only three rules, which makes their importance from a practical

point of view rather low. Nevertheless, their existence is justified by the following theorem that stems from them. This theorem already provides us with a general result for an arbitrary number of fuzzy rules.

Theorem 3.2.3

Let $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ be a linguistic description (1.3.7) and let $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. If there exists no rule $\mathcal{R}_k \in LD$ such that either (1)-(3a), (1)-(3b) or (4)-(6) holds, then $\mathcal{R}_i$ is redundant in LD.

PROOF: Let us denote

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : C_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0, w)\right) : C_2$$

where $LD' = LD \setminus \{\mathcal{R}_i\}$.

Let us prove the Theorem by contradiction, i.e., let us assume that although there exists no rule fulfilling either *(1)-(3a)*, *(1)-(3b)* or *(4)-(6)*, the rule $\mathcal{R}_i$ is not redundant in *LD*. This mean that for an arbitrary $w \in W$ there exists $u_0 \in w$ for which $C_1 \neq C_2$. This is possible only if there exists a rule $\mathcal{R}_p \in LD$ such that

$$\text{either} \quad (\alpha_p \to B_p) \in C_1 \quad \text{but} \quad (\alpha_p \to B_p) \notin C_2, \tag{3.2.1}$$

$$\text{or} \quad (\alpha_p \to B_p) \notin C_1 \quad \text{but} \quad (\alpha_p \to B_p) \in C_2 \tag{3.2.2}$$

where $\alpha_p = A_p(u_0)$.

There exist the following two possibilities for the rule $\mathcal{R}_p$:

$$\text{either} \quad (a)\ p \neq i \quad \text{or} \quad (b)\ p = i.$$

Let us discuss the case denoted by (3.2.1). It occurs when

$$A_p \in LPerc^{LD}(u_0, w) \quad \text{but} \quad A_p \notin LPerc^{LD'}(u_0, w) \tag{3.2.3}$$

Let us investigate both possibilities (a), (b).

(a) $p \neq i$

Then due to (3.2.3): $A_p \in LPerc^{LD}(u_0, w)$ and therefore, $A_p(u_0) \geq A_h(u_0)$ for arbitrary $\mathcal{R}_h \in LD$ from which necessarily $A_p \in LPerc^{LD'}(u_0, w)$ so, (a) contradicts (3.2.3) and it cannot occur when assuming (3.2.1).

(b) $p = i$

In this case, $A_i \notin LPerc^{LD'}(u_0, w)$ due to the fact that $\mathcal{R}_i \notin LD'$. There has to exist $\mathcal{R}_k \in LD'$ such that

$$A_k \in LPerc^{LD'}(u_0, w) \tag{3.2.4}$$

and assuming that $\mathcal{R}_i$ is not redundant we get $\mathcal{B}_k \neq \mathcal{B}_i$. Hence, the property denoted as *(1)* is necessary if $\mathcal{R}_i$ is supposed to be not redundant by (3.2.1).

Because $\mathcal{B}_k \neq \mathcal{B}_i$, i.e., $\mathcal{B}_k \neq \mathcal{B}_j$ (due to the fact that $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ and thus $\mathcal{B}_i = \mathcal{B}_j$), then clearly $\mathcal{R}_k \neq \mathcal{R}_j$. There are three possible mutual orders of the linguistic expressions in the respective antecedents:

$$(I)\ \mathcal{A}_j \leq_{\mathrm{LE}} \mathcal{A}_k,\ (II)\ \mathcal{A}_j \parallel_{\mathrm{LE}} \mathcal{A}_k,\ (III)\ \mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_j.$$

Let us discuss the case $(I)\ \mathcal{A}_j \leq_{\mathrm{LE}} \mathcal{A}_k$.

In this case, property (3.2.4) is possible only if $A_k(u_0) > A_j(u_0)$ and consequently $A_k(u_0) > A_i(u_0)$ and therefore $A_i \notin LPerc^{LD}(u_0, w)$ which contradicts (b). Consequently, (b) and (I) do not occur together when assuming (3.2.1).

Analogously, even in case of $(II)\ \mathcal{A}_j \parallel_{\mathrm{LE}} \mathcal{A}_k$ property (3.2.4) leads to the same conclusion that $A_i \notin LPerc^{LD'}(u_0, w)$ which contradicts (b). Consequently, (b) and (II) do neither occur together when assuming (3.2.1).

Finally, the case $(III)\ \mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_j$, denoted as *(2)*, is the only possibility.

So far, we have proved that assuming (3.2.1) necessarily leads to an existence of a rule fulfilling conditions *(1), (2)*. Finally, let us discuss the mutual order of the linguistic expressions $\mathcal{A}_i$ and $\mathcal{A}_k$.

If $\mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_i$, then due to (b): $A_i \in LPerc^{LD}(u_0, w)$ necessarily $A_i(u_0) > A_k(u_0)$ but then also $A_j(u_0) > A_k(u_0)$ which leads to $A_k \notin LPerc^{LD'}(u_0, w)$ which contradicts (3.2.4). Then necessarily $\mathcal{A}_k \not\leq_{\mathrm{LE}} \mathcal{A}_i$ which is equivalent to *(3a)* or *(3b)*. Therefore, assuming (3.2.1) is in a contradiction with non-existence of $\mathcal{R}_k$ fulfilling *(1)-(3a)* or *(1)-(3b)*.

The other possibility of non-redundancy may be caused by (3.2.2). It occurs when

$$A_p \notin LPerc^{LD}(u_0, w) \text{ but } A_p \in LPerc^{LD'}(u_0, w). \tag{3.2.5}$$

Let us consider both possibilities (a) and(b), but only in the reverse order.

(b) $p = i$

This case is immediately impossible because it contradicts (3.2.5), which requires $A_i \in LPerc^{LD'}(u_0, w)$ which is not possible due to the fact that $\mathcal{R}_i \notin LD'$.

(a) $p \neq i$

In this case, we assume that there has to exist $\mathcal{R}_p \in LD'$ such that

$$A_p \in LPerc^{LD'}(u_0, w) \tag{3.2.6}$$

and assuming that $\mathcal{R}_i$ is not redundant we get $\mathcal{B}_p \neq \mathcal{B}_i$. So, the property denoted as *(4)* is necessary if $\mathcal{R}_i$ is supposed to be not redundant by (3.2.1).

Because $\mathcal{B}_p \neq \mathcal{B}_i$, clearly $\mathcal{R}_p \neq \mathcal{R}_j$. There are three possible mutual orders of linguistic expressions in their respective antecedents:

$$(I)\ \mathcal{A}_j \leq_{\mathrm{LE}} \mathcal{A}_p,\ (II)\ \mathcal{A}_p \parallel_{\mathrm{LE}} \mathcal{A}_j,\ (III)\ \mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_j.$$

Let us discuss the case (I) $\mathcal{A}_j \leq_{\mathrm{LE}} \mathcal{A}_p$.

In this case, property (3.2.6) is possible only if $A_p(u_0) > A_j(u_0)$ and consequently $A_p(u_0) > A_i(u_0)$ and therefore $A_p \in LPerc^{LD}(u_0, w)$ which contradicts (3.2.5). Consequently, (a) and (I) do not occur together when assuming (3.2.2).

Let us discuss the case (II) $\mathcal{A}_p \parallel_{\mathrm{LE}} \mathcal{A}_j$.

There are two clear subcases, either $\mathcal{A}_p$ and $\mathcal{A}_j$ are of the same type or not. If they were not of the same type, then necessarily (3.2.6) leads to $A_p(u_0) \geq A_j(u_0)$ and consequently to $A_p(u_0) \geq A_i(u_0)$. Since neither $\mathcal{A}_p$ and $\mathcal{A}_i$ can be of the same type, then $\mathcal{A}_i \not\leq_{\mathrm{LE}} \mathcal{A}_p$ and therefore $A_p \in LPerc^{LD}(u_0, w)$ which contradicts (3.2.5). Therefore, $\mathcal{A}_p$ and $\mathcal{A}_j$ of the same type is the only possibility for the case (II) when assuming (3.2.2), i.e, we get the condition denoted as *(5)*.

The last thing we involve into our consideration for the case (II) is the mutual order of $\mathcal{A}_i$ and $\mathcal{A}_p$. Obviously, $\mathcal{A}_i \not\leq_{\mathrm{LE}} \mathcal{A}_p$ leads to $A_p \in LPerc^{LD}(u_0, w)$ which again contradicts (3.2.5). So, assuming (3.2.2) either yields the existence of a rule $\mathcal{R}_p$ fulfilling properties *(4)-(6)* or the case (III) not investigated yet.

Let us finally discuss the case (III) $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_j$, which is denoted by *(2)*.

Property (3.2.6) together with (III) leads to $A_p(u_0) = A_j(u_0) = 1$ and consequently to $A_p(u_0) \geq A_i(u_0) = 1$ and if even $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_i$ then it leads to the conclusion that

$$A_p \in LPerc^{LD}(u_0, w)$$

which contradicts (3.2.5). Consequently, (a) and (III) can occur together when assuming (3.2.2) only if the condition $\mathcal{A}_p \not\leq_{\mathrm{LE}} \mathcal{A}_i$ equivalent to *(3a)* and *(3b)* holds.

So, we have proved that assuming (3.2.1) necessarily leads to an existence of a rule $\mathcal{R}_k \in LD$ fulfilling conditions *(1)-(3a)* or *(1)-(3b)* while assuming (3.2.2) necessarily leads to the existence of a rule $\mathcal{R}_p \in LD$ fulfilling either conditions *(1)-(3a)*, *(1)-(3b)* or *(4)-(6)*, which completes the proof. □

The main contribution of Theorem 3.2.3 is that it provides a full characterization of fuzzy rules that may cancel the suspicion of redundancy and that no other fuzzy rules may be responsible for that. Hence, we may introduce the notion of *cancelling rule*.

Definition 23 Let $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ be a linguistic description (1.3.7) and let $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. If either *(1)-(3a)*, *(1)-(3b)* or *(4)-(6)* holds for $\mathcal{R}_k \in LD$, then the rule $\mathcal{R}_k$ is called a *cancelling* rule.

Theorem 3.2.3 then actually states that if some suspected $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ exists and no cancelling rule exists in LD, then $\mathcal{R}_i$ is redundant.

3.3 Complete answer

Sections 3.1 and 3.2 introduced basic concepts of a redundant rule, a rule that is suspected of redundancy with respect to another rule and also the concept of cancelling rules. We obtained a full characterization of cancelling rules. Hence, we know in which situation a given suspicion may be canceled. However, we also know that even such cancellation may be also eliminated (by another rule) and the suspected rule may be really redundant even if there exists a cancelling rule, see Figure 3.3 and Figure 3.5.

In this section, we attempt to obtain a complete answer on a given question whether a rule is redundant in a given linguistic description or not.

Theorem 3.3.1

Let LD be a linguistic description, let $\{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\} \subseteq LD$ *and let* $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. *Furthermore, let either* $(1) - (3a)$ *or* $(4) - (6)$ *holds for* $\mathcal{R}_k$ *and no further cancelling rule related to* $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ *exists in LD. Then, it holds that if* $\mathcal{R}_i$ *is redundant in LD then there exists a rule* $\mathcal{R}_p \in LD$, $\mathcal{R}_p \neq \mathcal{R}_k$ *such that*

a) $\mathcal{R}_i \hookrightarrow \mathcal{R}_p$,

b) $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_k$.

PROOF: Let assume that $(1) - (3a)$ holds for $\mathcal{R}_k \in LD$. Let us denote

$$r_{PbLD}\left(LPerc^{LD}(u_0, w)\right) : C_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0, w)\right) : C_2,$$

where $LD' = LD \setminus \{\mathcal{R}_i\}$. Let $\mathcal{R}_i$ be redundant in LD. It means that $C_1 = C_2$ for each $w \in W$ and for each $u_0 \in w$. Due to Lemma 1.3.1, for each $w \in W$ there exists $u_0 \in w$ such that $LPerc^{LD}(u_0, w) = \{A_i\}$ and thus $C_1 = \{B_i\}$, for such $u_0 \in w$. However, since $\mathcal{R}_i \notin LD'$, obviously $B_i \notin C_2$ for such $u_0 \in w$. Furthermore, because $\mathcal{R}_i$ is redundant, C_1 and C_2 have to be equal and thus there has to exist $\mathcal{R}_p \in LD$ such that $C_2 = \{B_p\}$ and

$$\mathcal{B}_p = \mathcal{B}_i \tag{3.3.1}$$

and moreover that $LPerc^{LD'}(u_0, w) = \{A_p\}$. This may happen only if $A_p(u_0) = 1$ because also $A_k(u_0) = 1$ and therefore necessarily

$$\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_k. \tag{3.3.2}$$

Since $A_p \notin LPerc^{LD}(u_0, w)$ the following also has to hold:

$$\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_p. \tag{3.3.3}$$

Formulas (3.3.1), (3.3.2) and (3.3.3) prove the Theorem for the assumption that $(1)-(3a)$ holds for $\mathcal{R}_k$. The proof for the assumption that $(4)-(6)$ holds for $\mathcal{R}_k$ is analogous. Hence, it is omitted. □

The importance of Theorem 3.3.1 for the procedure that determines redundant rules in a given linguistic description is following. This proves that in the case we have a cancelling rule fulfilling $(1)-(3a)$ or $(4)-(6)$, we do not have to investigate this pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ anymore because the influence of the cancelling rule may be *eliminated* only by another rule $\mathcal{R}_p$ with respect to which $\mathcal{R}_i$ is suspected of being redundant and moreover, the cancelling rule does not have the cancellation property with respect to this "*eliminating*" rule $\mathcal{R}_p$. Thus, in order to detect the redundancy of $\mathcal{R}_i$ it is sufficient to investigate this new suspected rule $\mathcal{R}_i \hookrightarrow \mathcal{R}_p$. Once more we refer to Figure 3.3 and Figure 3.5 that depict these situations.

Finally, we should investigate also the case of a cancelling rule fulfilling properties $(1)-(3b)$. This is the most complicated case, because, as we will show below, the elimination is not always achieved based on a rule for which the investigated rule $\mathcal{R}_i$ is also suspected. However, a satisfactory answer is even obtained in this case.

Theorem 3.3.2

Let LD be a linguistic description, let $\{\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k\} \subseteq LD$ *and let* $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$. *Furthermore, let* $(1)-(3b)$ *holds for* $\mathcal{R}_k$ *and no further cancelling rule related to* $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ *exists in LD. Then, it holds that* $\mathcal{R}_i$ *is redundant in LD if and only if there exists a rule* $\mathcal{R}_p \in LD$, $\mathcal{R}_p \neq \mathcal{R}_k$ *such that*

a) $\mathcal{A}_k \not\leq_{\mathrm{LE}} \mathcal{A}_p$,

b) $\mathrm{Ker}(A_i) \cap \mathrm{Ker}(A_k) \subseteq \mathrm{Ker}(A_p)$

c) $\mathcal{B}_p = \mathcal{B}_i$ *or* $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_i$

PROOF: Let us denote

$$r_{PbLD}\left(LPerc^{LD}(u_0,w)\right):C_1,$$
$$r_{PbLD}\left(LPerc^{LD'}(u_0,w)\right):C_2,$$

where $LD' = LD \setminus \{\mathcal{R}_i\}$. Let $\mathcal{R}_i$ be redundant in LD, i.e. $C_1 = C_2$.

Because $\mathcal{A}_i \parallel_{\mathrm{LE}} \mathcal{A}_k$, for any $w \in W$ there exists $u_0 \in w$ such that

$$A_i(u_0) = 1 \text{ and } A_k(u_0) = 1. \tag{3.3.4}$$

Let us assume that there exists no $\mathcal{R}_p \in LD$ such that it fulfills *a)*. It means that either no $\mathcal{R}_p \in LD$, $\mathcal{R}_p \neq \mathcal{R}_k$ exists at all which would directly lead to a contradiction with the assumption on redundancy of $\mathcal{R}_i$ or for any $\mathcal{R}_p \in LD$ it holds that $\mathcal{A}_k \leq_{\mathrm{LE}} \mathcal{A}_p$. A direct consequence of the latter case is that also $A_p(u_0) = 1$ for the given $u_0 \in w$. But due to the order of the antecedents we get

$$LPerc^{LD}(u_0,w) = \{A_i, A_k\} \quad \text{and} \quad LPerc^{LD'}(u_0,w) = \{A_k\} \tag{3.3.5}$$

and consequently

$$C_1 = \{B_i, B_k\} \quad \text{and} \quad C_2 = \{B_k\} \tag{3.3.6}$$

which due to $\mathcal{B}_i \neq \mathcal{B}_k$ contradicts the assumption that $C_1 = C_2$.

Hence, necessarily there has to exist at least one rule $\mathcal{R}_p \in LD$, $\mathcal{R}_p \neq \mathcal{R}_k$ such that *a)* holds but let us assume that none of these rules fulfills *b)*. Then, for any $w \in W$ there exists $u_0 \in w$ such that (3.3.4) holds but $A_p(u_0) < 1$ for such u_0 and thus, (3.3.5) and consequently (3.3.6) again holds which is again in a contradiction with the assumption that $C_1 = C_2$. It follows that necessarily there has to exist at least one rule $\mathcal{R}_p \in LD$, $\mathcal{R}_p \neq \mathcal{R}_k$ such that *a)-b)* hold.

There are only the following four possible mutual positions of antecedents of such a rule $\mathcal{R}_p$ and rules $\mathcal{R}_k, \mathcal{R}_i$:

(I) $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_k$ and $\mathcal{A}_i \parallel_{\mathrm{LE}} \mathcal{A}_p$,

(II) $\mathcal{A}_p \parallel_{\mathrm{LE}} \mathcal{A}_k$ and $\mathcal{A}_i \parallel_{\mathrm{LE}} \mathcal{A}_p$,

(III) $\mathcal{A}_p \parallel_{\mathrm{LE}} \mathcal{A}_k$ and $\mathcal{A}_i \leq_{\mathrm{LE}} \mathcal{A}_p$,

(IV) $\mathcal{A}_k \not\leq_{\mathrm{LE}} \mathcal{A}_p$ and $\mathcal{A}_p \leq_{\mathrm{LE}} \mathcal{A}_i$.

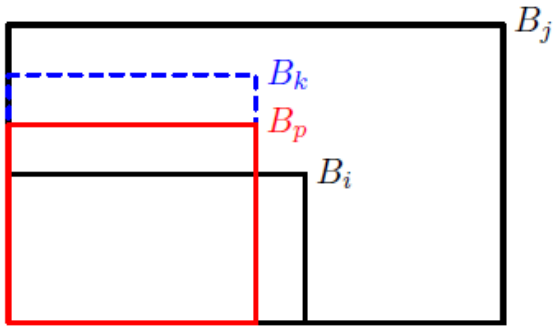


Figure 3.6: Situation for case (I) from the proof of Theorem 3.3.2.

In the case (I), that is displayed on Figure 3.6, for each $w \in W$ there exists $u_0 \in w$ such that the perception function gives the following results

$$LPerc^{LD}(u_0, w) = \{A_p, A_i\} \quad \text{and} \quad LPerc^{LD'}(u_0, w) = \{A_p\}.$$

No other fuzzy set may be an element of the result of the perception function because of the assumption that no other cancelling rule occurs with respect to the suspected rule $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$.

Such a result of the perception function leads to

$$C_1 = \{B_p, B_i\}, C_2 = \{B_p\}$$

which means that $C_1 = C_2$ if and only if $\mathcal{B}_p = \mathcal{B}_i$.

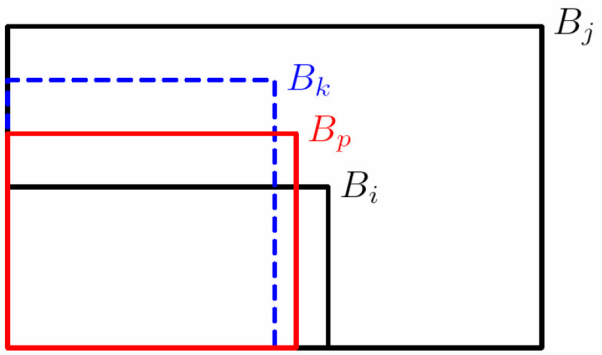


Figure 3.7: Situation for case (II) from the proof of Theorem 3.3.2.

Similarly, in the case (II), that is displayed on Figure 3.7, for each $w \in W$ there exists $u_0 \in w$ such that the perception function gives the following results

$$LPerc^{LD}(u_0, w) = \{A_p, A_i, A_k\} \quad \text{and} \quad LPerc^{LD}(u_0, w) = \{A_p, A_k\}$$

which leads to

$$C_1 = \{B_p, B_i, B_k\} \quad \text{and} \quad C_2 = \{B_p, B_k\}$$

which means that $C_1 = C_2$ if and only if $\mathcal{B}_p = \mathcal{B}_i$.

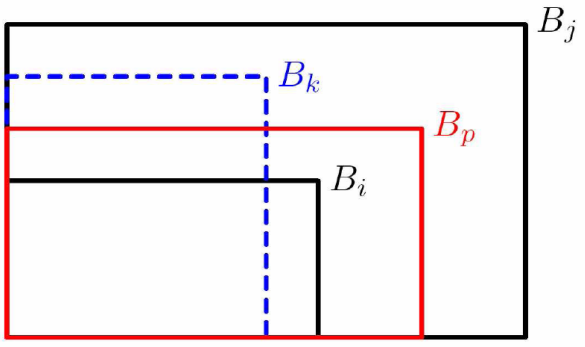


Figure 3.8: Situation for case (III) from the proof of Theorem 3.3.2.

Once again, in the case (III), that is displayed on Figure 3.8, for each $w \in W$

there exists $u_0 \in w$ such that the perception function gives the following results

$$LPerc^{LD}(u_0, w) = \{A_i, A_k\} \quad \text{and} \quad LPerc^{LD'}(u_0, w) = \{A_p, A_k\}$$

which leads to

$$C_1 = \{B_i, B_k\} \quad \text{and} \quad C_2 = \{B_p, B_k\}$$

which means that $C_1 = C_2$ if and only if $\mathcal{B}_p = \mathcal{B}_i$.

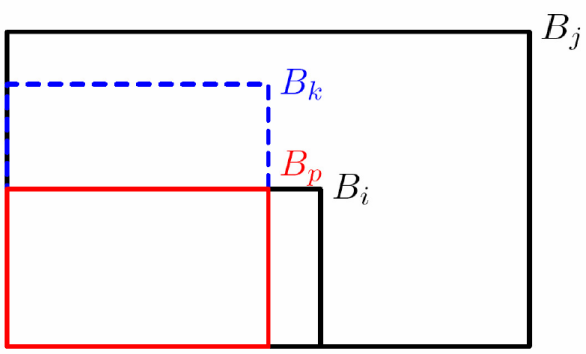


Figure 3.9: Situation for case (IV) from the proof of Theorem 3.3.2.

Finally, in the case (IV), that is displayed on Figure 3.9, for each $w \in W$ and for each $u_0 \in w$ for which *b)* holds (this is the investigated area), the perception function gives the following results

$$LPerc^{LD}(u_0, w) = \{A_p, A_k\} \quad \text{and} \quad LPerc^{LD'}(u_0, w) = \{A_p, A_k\}$$

which leads to $C_1 = C_2$ without any further requirements.

The cases (I)-(III) lead to $\mathcal{B}_p = \mathcal{B}_i$ and the case (IV) assumes $\mathcal{A}_p \leq_{\text{LE}} \mathcal{A}_i$ which together prove that *c)* has to hold for $\mathcal{R}_p$ and this completes the proof. □

3.4 Implementation and applications

The theoretical results introduced in Sections 3.1, 3.2 and 3.3 are important for applications because of two reasons. First, they indicate the potential directions

for the further redundancy analysis of other, possibly related, methods and models. Second, they may be employed directly to design an algorithm that searches for redundant rules in a given linguistic description. This algorithm has been designed and it is presented in this section. Furthermore, we describe a brief real-world demonstration of its performance.

Algorithm LDRed: `input` LD**,**

`1)` Preprocessing (using Remark 1)*).

`2)` Using Theorem 3.1.1, search for all pairs $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ in LD. Denote a set of Investigated Pairs as IP.

`3a)` For a pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j \in IP$ search for an $\mathcal{R}_k \in LD$ (using Theorem 3.2.3).

`3b)` If there is no such $\mathcal{R}_k$, delete $\mathcal{R}_i$ from LD and delete all pairs containing $\mathcal{R}_i$ from IP.

`3c)` If there is such an $\mathcal{R}_k \in LD$ for which either $(1)-(3a)$ or $(4)-(6)$ holds, the pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ is deleted from IP (using Theorem 3.2.3).

`4)` Repeat step `3)` for all the pairs from IP.

`5a)` For each remaining pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j \in IP$, there must be an $\mathcal{R}_k \in LD$ for which $(1)-(3b)$ holds. If there is another cancelling rule related to this pair, delete the pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ from IP.

`5b)` If there is another cancelling rule related to this pair, search for an $\mathcal{R}_p \in LD$ that satisfies $(a)-(c)$ from Theorem 3.3.2. If there is such an $\mathcal{R}_p$, then delete $\mathcal{R}_i$ from LD and delete all pairs containing $\mathcal{R}_i$ from IP. Otherwise, delete only the pair $\mathcal{R}_i \hookrightarrow \mathcal{R}_j$ from IP.

*)We delete all trivially redundant rules with the same antecedents and fully overlapping consequents.

6) Repeat step 5) for all the pairs from *IP*.

In the following chapter, we describe an ensemble approach to time series forecasting based on a combination of several forecasting methods. The combination is defined as a weighted mean of the forecasts obtained by individual methods and the goal is to determine appropriate weights for individual methods using fuzzy rules. The weight given to each method for each time series prediction is determined with the help of a linguistic description using quantitative features of the given time series as antecedent variables. Thus, a single linguistic description must be identified for each individual method. This can be achieved with the help of linguistic associations mining, particularly with a fuzzy variant of the GUHA method.

However, the fuzzy GUHA method necessarily produces many approved but redundant IF-THEN rules, especially when it is apllied to enormous volumes of data. Obviously, an efficient method that significantly decreases the number of fuzzy rules in the generated linguistic descriptions is highly desirable.

As shown in the following chapter, the theoretical research that led to the design of the LDRed algorithm introduced in Section 3.3 significantly reduced the number of fuzzy rules in linguistic descriptions. Moreover, this reduction did not affect the behavior of the rule base compared with the original rule base.

We also provide the linguistic descriptions before and after the redundancy analysis (Table 3.1). The linguistic descriptions were derived from a real-world application related to ensemble time series forecasting.

For example, we can see that $\mathcal{R}_1 \hookrightarrow \mathcal{R}_7$. However, there is a cancelling rule $\mathcal{R}_6$ fulfilling $(1) - (3b)$ but no elimination rule. Nevertheless, $\mathcal{R}_1$ is redundant because $\mathcal{R}_1 \hookrightarrow \mathcal{R}_8$ also holds and there is no cancelling rule related to this suspected rule. Later, $\mathcal{R}_8$ was also deleted as redundant because of the suspected rule $\mathcal{R}_8 \hookrightarrow \mathcal{R}_7$

Rule	**IF part**		**THEN part**
	X_1	X_2	Y
$\boldsymbol{\mathcal{R}_1}$	**Me**	**Sm**	**Ro Bi**
$\boldsymbol{\mathcal{R}_2}$	**ML Me**	**Sm**	**Ro Bi**
$\mathcal{R}_3$	**ML Me**	**Ve Sm**	**Ro Bi**
$\mathcal{R}_4$	ML Sm	Ve Sm	Ro Bi
$\boldsymbol{\mathcal{R}_5}$	**Ro Me**	**Ex Sm**	**Ro Bi**
$\boldsymbol{\mathcal{R}_6}$	**Ro Me**	**Ex Sm**	**ML Bi**
$\mathcal{R}_7$	Ro Me	ML Sm	Ro Bi
$\boldsymbol{\mathcal{R}_8}$	**Ro Me**	**Sm**	**Ro Bi**
$\boldsymbol{\mathcal{R}_9}$	**Ro Me**	**Ve Sm**	**Ro Bi**
$\mathcal{R}_{10}$	Ro Me	Ve Sm	ML Bi
$\mathcal{R}_{11}$	Sm	Ro Me	Ro Bi
$\mathcal{R}_{12}$	Sm	Sm	Ro Bi
$\boldsymbol{\mathcal{R}_{13}}$	—	**Ex Sm**	**Ro Bi**
$\mathcal{R}_{14}$	—	**Ex Sm**	**ML Bi**
$\mathcal{R}_{15}$	—	Sm	Ro Bi
$\boldsymbol{\mathcal{R}_{16}}$	—	**Ve Sm**	**Ro Bi**
$\mathcal{R}_{17}$	—	Ve Sm	ML Bi

Table 3.1: Example of the performance of the proposed algorithm, LDRed. Redundant rules are denoted in bold, other rules remain in the description.

and there was no cancelling rule.

3.5 Conclusions

We introduced our approach for detecting so-called redundancies in systems of fuzzy/linguistic IF-THEN rules (linguistic descriptions). We showed that intuitively redundant rules are not always redundant from a formal point of view. Hence, we introduced a deeper and formally correct approach.

Our approach is based on detecting the rules that are suspected of redundancy and that require further investigation. We provided a full characterization of the rules that are suspected of redundancy. We also disproved intuitively valid hypotheses determining situations when a rule suspected of redundancy is not redundant. However, both hypotheses led us to a full characterization of "cancelling rules", i.e.,

to a full characterization of rules that may cancel those suspected of redundancy.

After considering the preliminary study, we focused on further investigation of situations that occur when cancelling rules exist. These situations may lead to actual confirmation of the cancellation of the suspicion, or determining that a suspected rule is indeed redundant. We introduced theoretical results that allowed us to propose a deterministic algorithm that could automatically check (and remove) redundancies in any linguistic description consisting of a finite number of fuzzy/linguistic IF-THEN rules. These theoretical results were formulated in the theorems provided in Sections 3.3, which also provided the requisite deterministic algorithm, LDRed. Section 3.4 demonstrated the need for such an algorithm and it illustrated the performance of the algorithm on a real application that used linguistic descriptions generated from data using the fuzzy GUHA method.

In general, when a linguistic description is higher-dimensional (i.e., when it consists of IF-THEN rules with more than one antecedent variable, see (1.3.11) and discussion above), it may seem that there are not many IF-THEN rules with antecedents that can be ordered using the ordering $\leq_{(u_0,w)}$ from Section **??**. Hence, it may seems that our method is not very useful in this case. However, if these IF-THEN rules are generated automatically from data by a learning procedure, redundant rules can occur quite often, irrespective of the dimension of the linguistic description. Thus, their detection and removal can be really useful from the point of view of performance and interpretability. However, if no rules are found that are suspected of redundancy, then we can say that there are no redundancies in the strict formal sense of Definition 21. In general, this formal understanding of redundancy, which stresses that original and new linguistic descriptions are equivalent from the point of view of their behavior, is significantly different from other approaches that mainly aim to simplify linguistic descriptions [109, 110] using various techniques,

such as merging rules. Of course, their use may be also beneficial. However, there is no guarantee that the output of simplified linguistic descriptions is equivalent to the original output. Finally, in the case of learning high-dimensional fuzzy/linguistic IF-THEN rules, we can restrict the number of linguistic hedges (Section **??**) and the number of rules may be reduced (because similar data records produce identical rules that are pruned down).

Chapter 4

On ensemble techniques for time series forecasting

4.1 Introduction and motivation

As mentioned in Section 1.1, there are many different methods to predict future values of a time series. Unfortunately, there is no single forecasting method that generally outperforms any other. Thus, there is a danger of choosing a method which is inappropriate for a given time series. Note that even searching for methods that outperform any other for narrower specific subsets of time series has not been successful yet, recall e.g., [6], where the authors stated:

"Although forecasting expertise can be found in the literature, these sources often fail to adequately describe conditions under which a method is expected to be successful".

4.1.1 Ensembles

In order to eliminate the risk of choosing an inappropriate method, distinct ensemble techniques (ensembles in short) have been designed and successfully applied. The

main idea of ensembles consists in an appropriate combination of several forecasting methods. Typically, the ensemble techniques are constructed as a linear combination of the individual ones. It can be described as follows. Let us assume that we are given a set of M individual methods and for a given times series $y_1, y_2, \ldots, y_T$ and a given forecasting horizon h, j-th individual method provides us with the following prediction:

$$\hat{y}_{T+1}^{(j)}, \hat{y}_{T+2}^{(j)}, \ldots, \hat{y}_{T+h}^{(j)}, \quad j = 1, \ldots, M.$$

Then the ensemble forecast is given by the following formula:

$$\hat{y}_{T+i} = \frac{1}{\sum_{j=1}^{M} w_j} \cdot \sum_{j=1}^{M} w_j \cdot \hat{y}_{T+i}^{(j)}, \quad i = 1, \ldots, h$$

where $w_j \in \mathbb{R}$ is a weight of the j-th individual method. These weights are usually normalized, that is $\sum_{j=1}^{M} w_j = 1$.

Let us recall that Bates and Granger [10] was one of the first to show significant gains in accuracy through combination. Another early work by Newbold and Granger [86] combined various time series forecasts and compared the combination against the performance of the individual methods. They showed that for a set of forecasts, a linear combination of these forecasts could be obtained which would also be unbiased and achieve a combined forecast error variance smaller than the individual forecasts. They found that the combining procedures produced an overall forecast superior to individual forecasts on the majority of tested time series.

How to combine methods, i.e., how to determine appropriate weights, is still a relatively open question. For instance, Makridakis et al. [80] showed that taking a simple average outperforms taking a weighted average method combination. In other words, the so-called "equal-weights combining" [23], that is an arithmetic mean of individual forecasts, is a benchmark that is hard to beat and finding appropriate non-equal weights rather leads to a random damage of the main averaging idea that

is behind the robustness and accuracy improvements.

4.1.2 Motivation for the suggested approach

Although the equal-weights performs as accurately as mentioned above, there are works that promisingly show the potential of more sophisticated approaches. We recall Lemke and Gabrys [75] that describes an approach using meta-learning for time series forecasting based on the features of time series such as: a measure for the strength of the trend, the standard deviation, the skewness, etc. Given time series were clustered using the k-means algorithm. Individual methods were ranked according to their performance on each cluster and then the three best methods for each cluster were selected. For a given new time series, the closest cluster was determined and the given three best methods were combined.

It should be stressed that this approach performed very well on sufficiently big set of time series. For us, this approach is one of the main motivations because it demonstrates that there exists a dependence between time series features and a performance of a forecasting method.

The second major motivation stems from the so-called *Rule-Based Forecasting* developed by Collopy and Armstrong [6, 23]. It is an expert system that uses domain knowledge to combine forecasts from various forecasting methods. Using IF-THEN rules, the Rule-Based Forecasting determines the weights of individual forecasting methods.

We follow the main ideas of the rule-based forecasting [6] and of using time series features [75] to obtain an interpretable and understandable model.

4.2 Fuzzy Rule-Based Ensemble

As mentioned above, the Rule-Based Forecasting uses the rules to determine weights [6]. However, only few of these rules are directly used to set up weights. Most of them set up rather a specific model parameters, e.g., the smoothing factors of the Brown's exponential smoothing with trend. Moreover, in antecedents, the rules very often use properties that are not crisp but rather vague, e.g., expressions such as: "last observation is unusual; trend has been changing; unstable recent trend" etc., see [23]. For such cases, using crisp rules that are either fired or not and nothing between, seems to be less natural than using fuzzy rules. Similarly, the use of crisp consequents such as: "add 10% to the weight; subtract 0.4 from beta; add 0.1 to alpha" etc. [23], seems to be less intuitive than using vague expressions that are typical for fuzzy rules.

4.2.1 General structure of the model

Therefore, our goal was to propose a method that uses fuzzy rules instead of crisp rules in order to capture the omnipresent vagueness in the expressions; to use only quantitative features (no domain knowledge) in the antecedent variables which enable to fully automatize the method; to use only individual forecasting method weights as the consequent variables [112, 113]. The result of such motivated investigation is the *Fuzzy Rule-Based Ensemble* that is schematically illustrated on Figure 4.1.

The Fuzzy Rule-Based Ensemble method uses a single linguistic description, i.e. a fuzzy rule base with evaluative linguistic expressions [89], for each forecasting method. Each of these linguistic descriptions determines a weight of a single individual method based on transparent and interpretable rules, such as:

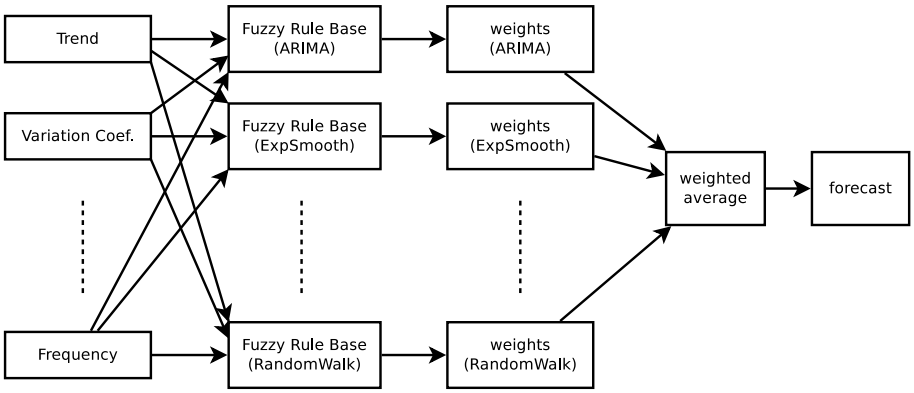


Figure 4.1: Structure of the Fuzzy Rule-Based Ensemble method.

"**IF** *Strength of Seasonality is Small* **AND** *Coefficient of Variation is Roughly Small* **THEN** *Weight of the j-th method is Big.*"

After an appropriate inference method is applied (see Section 4.2.2) in order to obtain a fuzzy output, a defuzzification method is employed and thus, a crisp result (weight of a particular method) is determined.

So far, based on experiments and previous publications [75], the following features were considered: *strength of trend, strength of seasonality, length of the time series, skewness, kurtosis, coefficient of variation, stationarity and frequency.*

Based on listed features, the inference mechanism sets weights to the following forecasting methods in our ensemble: *seasonal Autoregressive Integrated Moving Average* (ARIMA), *Exponential Smoothing*, *Theta* and the *Random Walk process.* We denote defuzzified values of these weights as $w_{AR}, w_{ES}, w_{Theta}$, and w_{RW}, respectively.

4.2.2 Components of the model

In order to estimate (set up) a particular value of the weight for each forecasting method with help of the fuzzy rules, an appropriate fuzzy inference mechanism has to be employed. As mentioned above, the Fuzzy Rule-Based Ensemble method employs linguistic descriptions, i.e., the fuzzy rule bases with so-called evaluative linguistic expressions.

Such linguistic expressions have their theoretical model of the semantics based on intension, context and extension, which is described in 1.3.1.

If a fuzzy rule base is viewed as a linguistic description and thus, uses the above recalled evaluative linguistic expressions with their model of semantics, one can neither model the rules (and consequently the whole description) as a conjunction of implicative rules nor as a disjunction of conjunctions (Mamdani-Assilian model). The used expressions, mainly their full overlapping, require a specific inference method – *Perception-based Logical Deduction*, see 1.3.3.

Finally, the inferred output is defuzzified by the Defuzzification of Evaluative Expressions (DEE) that has been designed specifically for the outputs of the PbLD inference mechanism. In the case of the Fuzzy Rule-Based Ensemble method, the defuzzification DEE is applied after the inference so that, the deduced weights $w_{AR}, w_{ES}, w_{Theta}, w_{RW}$ displayed on Figure 4.1 are already crisp numbers.

4.2.3 Fuzzy rule base identification

The last missing point is the identification of the linguistic descriptions. This may be done by distinct approaches. One could expect a deep applicable expert knowledge, however, neither our experience nor the experience of others confirms this expectations. Let us once more refer to the observation of Armstrong, Collopy and

Adya in [6], already recalled in Section 4.1.

Because of the missing reliable expert knowledge, we focus on data-driven approaches that may bring us the interpretable knowledge hidden in the data.

However, before we apply any data-mining technique, we have to clarify how we interpret the weights in the data because only the features serving as antecedent variables are measured. Naturally, the individual method weights should be proportionally higher if a given method is supposed to provide lower forecasting error and vice-versa. Thus, it is natural to put

$$w_j = 1 - acc_j \tag{4.2.1}$$

where acc_j denotes an appropriate normalized forecasting error of the j-th method. Now, any appropriate data-mining technique may be applied in order to determine the dependence between features and the precision (weight) of each method.

4.3 Generating fuzzy rules bases

4.3.1 Application of fuzzy GUHA to Fuzzy Rule-Based Ensemble

As mentioned in Subsection (4.2.3), an appropriate data-driven approach may be used for the identification of the linguistic descriptions in the Fuzzy Rule-Based Ensemble. The chosen data-mining technique, in our case the fuzzy GUHA, may be applied on a time series data set in order to determine linguistic descriptions capturing the relationship between time series features and the forecasting accuracy. In this subsection, we describe only the main idea. Further details about the data sets and the features are described in Section (4.6).

Assuming we have a sufficiently big data set of times series, we may separate it

into a training set and a testing set. Then we determine features for every single time series from the training set and perform forecasts by all individual methods at disposal and we determine their accuracies for every single time series. Using (4.2.1), we determine the weight values for every single time series and each individual forecasting method. For each individual forecasting method, this approach transforms the training data set into a table similar to Table 4.1 which shows the case for the ARIMA method.

	Φ_1^{ExSm}	...	Φ_q^{ExBi}	$W_{\text{AR}}^{\text{ExSm}}$	...	$W_{\text{AR}}^{\text{ExBi}}$
TS_1	0.9	...	0.7	0	...	0.9
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
TS_n	0.1	...	0.2	0.8	...	0

Table 4.1: The transformed training data set for the ARIMA forecasting method.

Objects $\text{TS}_1, \ldots, \text{TS}_n$ in Table 4.1 are the time series from the training set; $\Phi_1, \ldots, \Phi_q$ are features of given time series; and symbol W_{AR} stands for the weight (inverted accuracy) of the ARIMA method.

The fuzzy GUHA then combinatorically generates hypotheses that are immediately statistically either declined or confirmed as linguistic associations based on the chosen quantifier parameters, see Example 5. These associations serve as linguistic descriptions that are used in Fuzzy Rule-Based Ensemble in order to determine the weight of each individual method in the ensemble for any time series. As we deal with four individual methods in our case, this led to the four-fold use of the method. The performance of such approach to build Fuzzy Rule-Based Ensemble is confirmed on the testing set, see Section 4.6.

Example 5 *Let* Φ_1 *be Length and* Φ_2 *be Kurtosis. Our fuzzy GUHA approach provided us with the following implicative hypothesis:*

$$C(Length^{\text{QRSm}}, \text{Kurtosis}^{\text{RoMe}}) \sqsubset_r^{\gamma} D(W_{\text{AR}}^{\text{QRBi}})$$

where Length and Kurtosis denote the length and the kurtosis of a given time series, respectively, and W_{AR} denotes the weight of the ARIMA method. This association was confirmed on the following confidence and support degree:

$$\gamma = 0.50, r = 0.09,$$

respectively.

Such a confirmed association may be viewed and thus, directly interpreted, as the following fuzzy rule:

"IF *Length is Quite Roughly Small* AND *Kurtosis is Roughly Medium* THEN *Weight of the ARIMA method is Quite Roughly Big.*"

4.4 Quality measures and the size reduction

In order to construct a "good" rule base, some measures of quality of the rules have to be employed. An extensive research has been made already especially with respect to crisp association rules. See e.g., [42] for a comprehensive survey on that topic.

Regarding fuzzy association rules, e.g., *fuzzy confirmation measures* have been introduced [44] and thoroughly studied [74, 45, 43]. Other works focus on the GUHA method [73, 48, 50] and its generalized quantifiers [108, 74] or quality measures based on the assumption of independence [17].

Whereas the studied measures focus on quality of a *single rule* in the rule base, in this section we describe an approach that employs the quality measure of the *whole* rule base as it has found useful in recent research [19].

4.4.1 The coverage of data

Due to the *curse of dimensionality* phenomenon, the number of tested hypotheses and also the number of confirmed associations may turn to be enormous. This is even strengthened by the use of evaluative linguistic expressions that are modeled by fuzzy sets in full inclusion, see Figure 1.3. An appropriate setting of confidence and support degrees is useful but usually not sufficient and some redundancy removal or simplification and size reduction algorithms are necessary to be used. In both, another global "quality measure" or concept, that will be defined below, may be very useful. The concept is called *the coverage of data by LD* and expresses a support of antecedents of a set of rules (associations).

Definition 24 Let us be given a set of objects $O = \{o_1, \ldots, o_n\}$ where $o_i = (o_{i1}, \ldots, o_{ik})$, let us be given a set of association rules of the type (1.5.1) represented as an $LD = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$. Let the value a_i^j for the rule $\mathcal{R}_j \in LD$ be given as follows:

$$a_i^j = A_{i1}^j(o_{i1}) \wedge \cdots \wedge A_{ik}^j(o_{ik})$$

where $A_{i1}^j, \ldots, A_{ik}^j$ denote the antecedent fuzzy sets of $\mathcal{R}_j$.

Then the *coverage of O by LD*, denoted by $\mathrm{cov}_{LD}(O) \in [0,1]$, is given as follows

$$\mathrm{cov}_{LD}(O) = \frac{\sum_{i=1}^{n} \bigvee_{j=1}^{m} a_i^j}{n} \ . \tag{4.4.1}$$

Remark 6 *Note that the definition of coverage appears already in [53] where the same idea is introduced. However, similarly to other quality measures, it relates only to a single rule, not to a set of rules. Moreover, that definition is provided for crisp rules only.*

The definition of coverage by (4.4.1) expresses a very intuitive measure of coverage of data by the generated rules. It is not necessarily expected to be close to 1, as

some data sets may be generated rather by a pure noise than by some statistically significant dependencies expressible by associations. On the other hand, high values of coverage clearly express high coverage of the data by generated association rules. Therefore, the concept of coverage may be very useful in setting the appropriate parameters of support and confidence degrees. Moreover, when applying any simplification or size reduction algorithm that modifies original yet too big LD into a smaller reduced LD', observing the difference between $\mathrm{cov}_{LD}(O)$ and $\mathrm{cov}_{LD'}(O)$ is highly desirable.

The problem of possible redundancy of fuzzy rules is well-known and has been studied by many authors, see [9, 109, 110]. Investigations presented so far provided mainly algorithms that may slightly change the output of modified rule bases in comparison to original ones, but are very efficient in size-reducing simplification. It is worth recalling, e.g., the study [41] that focuses on redundancies in Takagi-Sugeno fuzzy rules or another interesting investigation dealing with the redundancy in Takagi-Sugeno models [79] that is based on merging of similar rules. In Chapter 3, we focused on a theoretically-based algorithm of detection and removal of redundant rules in linguistic descriptions connected to the perception-based logical deduction. However, as we may see from Table 4.2, although the number of detected redundant rules in our application was very high, still the remaining size of the linguistic descriptions did not allow us to view the model as a transparent and interpretable white-box.

4.4.2 The size reduction algorithm

For the reason mentioned above, it is necessary to develop a size-reducing simplification algorithm also for linguistic descriptions. Apart from keeping $\mathrm{cov}_{LD'}(O)$ as close to $\mathrm{cov}_{LD}(O)$ as possible, the algorithm should possess the desirable property

of the least possible modification of the resulting function.

Natural yet partly naive approaches would harm this goal. For example, an algorithm taking k rules with the highest confidence would not be very appropriate as the highest confidence is always obtained for rules with narrow antecedents and very wide consequents. Thus, such modified LD' would not necessarily keep $\mathrm{cov}_{LD'}(O)$ as close to $\mathrm{cov}_{LD}(O)$ as possible, and moreover, the inferred weights would be very close to the equal weights. The reason is that the wider the consequent, the closer the defuzzified output is to the middle of the output universe. A similar approach based on k rules with the highest support would not be appropriate as there is always a problem of the determination of the parameter k.

One could design an algorithm that takes the rule from LD with the highest support to LD', and then it iteratively adds rules that increase $\mathrm{cov}_{LD'}(O)$ as much as possible up to a certain *reduction threshold*

$$\varrho = \frac{\mathrm{cov}_{LD'}(O)}{\mathrm{cov}_{LD}(O)}, \quad \varrho \in [0,1],$$

is exceeded, e.g., up to $\varrho \geq 0.9$. Setting the ϱ is much easier as this does not require any knowledge on the number of rules and one transparently expresses the lowest allowed decrease in the value of the coverage.

On the other hand, even such approach is not appropriate as it happens that, e.g., the following rule

$$\mathcal{R}' := \mathsf{IF}\ X_{i1}\ \text{is}\ \texttt{any}\ \mathsf{AND}\ \cdots\ \mathsf{AND}\ \mathtt{X_{im}}\ \text{is}\ \texttt{any}\ \mathsf{THEN}\ \mathtt{W}\ \text{is QR Bi}$$

with an "empty" antecedent *) is mined, as the precision of the given forecasting method may be at least quite roughly high for any time series independently on its features. Due to the empty antecedent, such rule fires everywhere, unless

*) Recall that the model of expression `any` attains normality at any point.

there are other rules with narrower antecedents. The influence of such rule is obviously positive, as it sets-up the default value of the weight quite roughly high, and not medium, which is statistically confirmed by GUHA as reasonable. However, for narrower antecedents, GUHA may mine other rules that determine even a higher weight, and obviously, some of these refining rules should be preserved by any appropriate reduction. Nevertheless, due to its empty antecedent, the linguistic description $LD' = \{\mathcal{R}'\}$ containing a single item has the coverage equal to 1, and therefore, any reduction threshold would be exceeded. So, such an algorithm would stop adding rules to LD' immediately and the reduced linguistic description would contain only $\mathcal{R}'$. As a consequence, all rules with other consequents would be omitted, which is not desirable anymore.

Therefore, in order to keep the modified LD' as varied as possible, first of all, we separate the mined LD into several sub-descriptions with the same consequent, and then, we apply the above mentioned algorithm separately to each of the sub-description. Finally, reduced sub-descriptions are merged into the final LD'. Formally, the algorithm may be described as follows, see Algorithm 1.

The process of the creation of rule bases for our ensemble of forecasts consists in an automatic detection and deletion of redundant rules based on a rather complicated and sophisticated, yet fully theoretically justified algorithm, see 3. As can be seen in Table 4.2, the redundancy detection algorithm reduced significantly the number of rules, although not sufficiently in some cases. After that, a heuristic size reduction and simplification algorithm was applied again on redundancy-free rule bases to obtain even smaller rule bases. For results see Table 4.2.

The reduction of rule bases is also beneficial from the perspective of the computational efficiency. Whereas the inference performed on a non-reduced rule base

Algorithm 1 Reduction of LD to LD'

$LD' \leftarrow \emptyset$
for each unique consequent $\mathcal{B}$ of any rule from LD **do**
    $LD_{\mathcal{B}} \leftarrow \{\mathcal{R} \in LD \mid \mathcal{B}$ is the consequent of $\mathcal{R}\}$
    $LD'_{\mathcal{B}} \leftarrow \emptyset$
    while $\mathrm{cov}_{LD'_{\mathcal{B}}}(o) < \mathrm{cov}_{LD_{\mathcal{B}}}(o) \cdot \varrho$ **do**
        Let $\mathcal{R} \in (LD_{\mathcal{B}} - LD'_{\mathcal{B}})$ be such rule that $\mathrm{cov}_{LD'_{\mathcal{B}} \cup \{\mathcal{R}\}}(o)$ is maximal
        $LD'_{\mathcal{B}} \leftarrow LD'_{\mathcal{B}} \cup \{\mathcal{R}\}$
    end while
    $LD' \leftarrow LD' \cup LD'_{\mathcal{B}}$
end for

Setting	Number of rules				Error	
	R-ARIMA	R-ES	R-THETA	R-RW	Avg.	SD
Fuzzy GUHA	807	2518	2380	4937	13.24	14.22
Redundancy Removal	204	1911	1261	2903	13.24	14.22
Reduction $\varrho = 1$	74	111	133	181	13.34	14.43
Reduction $\varrho = 0.975$	47	56	63	80	13.31	14.39
Reduction $\varrho = 0.95$	39	44	51	60	**13.29**	**14.37**
Reduction $\varrho = 0.925$	36	39	45	46	13.26	14.40
Reduction $\varrho = 0.9$	30	36	39	40	13.24	14.38
Reduction $\varrho = 0.85$	27	29	31	31	13.24	14.40
Reduction $\varrho = 0.8$	25	26	26	25	13.24	14.41
Reduction $\varrho = 0.7$	20	19	20	18	13.31	14.67

Table 4.2: Results of the FRBE method with fixed settings of the minimum support $r = 0.05$, the minimum confidence $\gamma = 0.5$ and various settings of the reduction threshold ϱ. The table shows the mean and standard deviation of SMAPE computed from forecasts of the time series from the testing data set. The variant selected by the cross-validation is in bold.

took hours for the whole data set, the reduced rule bases costed seconds of the computational time for the inference engine to prepare weights of each of the 2829 time series in the data set. See [19] for a more detailed study on that topic.

4.5 Implementation

4.5.1 Time series data sets and accuracy measures

To develop and validate the model, we have used 2829 time series from the M3 data set repository that contains 3003 time series from the M3-Competition [81]. The original M3 data set contains time series with yearly, quarterly, monthly, and unknown frequencies, the 74 time series with unknown frequencies were omitted.

Note that we talk about frequency in a sense of a periodicity of measurements and this feature is neither dependent on the seasonality, which can (but does not have to) be present or which can be multiple, nor on the periodicity of the data-generating process in the mathematical sense. Vast majority (nearly all) of real life situations lead to a time series where the frequency of measurements is at disposal and also vast majority of the M3 time series was provided by this feature. Therefore, we do find this feature very suitable and not restrictive.

Note also that omitting a time series with unknown measure frequency does not mean that our model cannot forecast time series with other frequencies or without the knowledge of the frequency. In such cases, this feature as an input variable is ignored and the ensemble combines the individual forecasting methods dependently only on the other features. Moreover, if there was a similar huge data set of time series with other frequencies at disposal, it would be only a matter of retraining the ensemble for these newly added frequencies. Similarly, the Fuzzy Rule-Based Ensemble method can be retrained to forecast some specific time series from a particular domain, and thus, one has to view this investigation showing a potential in a general perspective.

The M3 data set of time series serves as a generally accepted benchmark database provided by the authority of the International Institute of Forecasters. The time

series are of five categories: Microeconomy, Macroeconomy, Industry, Finance, Demography. The data set was divided into two distinct sets simply by putting time series with even or odd IDs into a *training* or *testing set*, respectively, see Table 4.3. The same table also indicates the distribution of time series lengths across the training and testing set. The lengths h of forecasting horizons (i.e. the number of values in the future to be forecasted) were the same as in the original M3-Competition: 18 for monthly, 8 for quarterly, and 6 for yearly time series.

Source	Training (Testing) Set			Total
	Monthly $h = 18$	Quarterly $h = 8$	Yearly $h = 6$	
By category:				
– Demographic	55 (56)	28 (29)	123 (122)	206 (207)
– Finance	73 (72)	38 (38)	29 (29)	140 (139)
– Industry	167 (167)	42 (41)	51 (51)	260 (259)
– Macro	156 (156)	168 (168)	41 (42)	365 (366)
– Micro	237 (237)	102 (102)	73 (73)	412 (412)
– Other	26 (26)	—	5 (6)	31 (32)
By length:				
– to 25	—	26 (26)	220 (227)	246 (253)
– 26–50	—	147 (150)	102 (96)	249 (246)
– 51–75	172 (167)	205 (202)	—	377 (369)
– 76–125	43 (48)	—	—	43 (48)
– 126+	499 (499)	—	—	499 (499)
Total	714 (714)	378 (378)	322 (323)	1414 (1415)

Table 4.3: The split of the M3 data set into the training set and the testing set. The table shows the number of time series for different categories and lengths. Accordingly to the original M3-Competition, forecasting horizons h were set identically for both training and testing set, as indicated in the table.

The training set was used for an identification of our model, that is, for generation of our fuzzy rule base. The testing set was used for testing whether the determined knowledge encoded in the fuzzy rules works generally well also for time series "not seen" by the rule base generation algorithm.

All forecasts were computed using the R-project, version 3.1.2, and the package `forecast` [64]. We started with the following forecasting methods that are available in that package: Autoregressive Integrated Moving Average (ARIMA), Exponential Smoothing, Random Walk, Random Walk with Drift, Theta model, Bats model, Linear model, Structural Time Series, and Forecasting using STL objects.

Using the *hill-climbing* optimization technique on the training data set, we selected such forecasting method that increased the accuracy of the whole ensemble maximally. After several steps, we ended with the following set of methods that formed our ensemble: *seasonal Autoregressive Integrated Moving Average* (abbr. R-ARIMA), *Exponential Smoothing* (R-ES), *Theta* (R-Theta) and the *Random Walk process* (R-RW). These methods were executed with a fully automatic parameter selection and optimization which made possible to concentrate the investigation purely on the combination technique. Moreover, their arithmetic mean (R-AM), that represents the equal weights ensemble method, was also determined and used as a valid benchmark.

To compare the forecasting methods, we use *Symmetric Mean Absolute Percentage Error* (SMAPE), see Subsection 1.1.3. Let SMAPE_{ij} represents a SMAPE value of the j-th individual method on the i-th training time series. Then the j-th method's weight w_{ij} for the i-th time series is computed accordingly to equation (4.2.1) in Subsection 4.2.3 as follows:

$$w_{ij} = 1 - acc_{ij},$$

where acc_{ij} is the SMAPE_{ij} normalized as follows:

$$acc_{ij} = \frac{\text{SMAPE}_{ij} - \min\{\text{SMAPE}_{.j}\}}{\max\{\text{SMAPE}_{.j}\} - \min\{\text{SMAPE}_{.j}\}}$$

with $\max\{\text{SMAPE}_{.j}\}$ and $\min\{\text{SMAPE}_{.j}\}$ being computed for each j over all time

series. That way is ensured that the weights of the individual methods are in the interval $[0, 1]$.

4.5.2 Time series features

For further investigation, some important features need to be extracted from a given time series. We used the following features: *frequency, length of the time series, absolute skewness, kurtosis, coefficient of variation, strength of trend, strength of seasonality and stationarity.*

Let a given time series $y_1, y_2, \ldots, y_T$ is of the frequency F, i.e., $F = 1, 4, 12$ for yearly, quarterly and monthly time series, respectively. Then, the features used to predict weights of the forecasting methods are defined as follows.

The *frequency* is given by the reciprocal value of F, i.e., it is given as $1/12$, $1/4$ and 1 in the case of the monthly, quarterly and yearly time series, respectively.

The *length of the time series* is equal to the number of known time lags.

The *absolute skewness* is given as

$$absolute\ skewness = \left| \frac{m_3}{{m_2}^{3/2}} \right|,$$

where $m_i = \frac{1}{T} \sum_{t=1}^{T} (y_t - \bar{y})^i$ and $\bar{y}$ is the arithmetic mean of the given time series $\{y_t\}_{t=1}^{T}$.

The *kurtosis* is given as

$$kurtosis = \frac{m_4}{{m_2}^2},$$

with m_i given as above.

The *coefficient of variation* is given as

$$CV = \frac{s_y}{\bar{y}},$$

where s_y is the standard deviation of $\{y_t\}_{t=1}^{T}$.

The *strength of trend* is given by $(1-p)$, where p is a p-value of a statistical test of a null hypothesis $H_0 : \beta_1 = 0$, where β_1 is a slope parameter of a linear regression model:

$$y_t = \beta_0 + \beta_1 t + \varepsilon_i,\ t \in \{1, 2, \ldots, T\}.$$

The *strength of seasonality* is given by $1 - \min\{p_2, p_3, \ldots, p_F\}$, where p_i (for $i \in \{2, 3, \ldots, F\}$) is a p-value of a test of a null hypothesis $H_0 : \beta_i = 0$, where β_i is a coefficient of the linear regression model

$$y_t = \beta_0 + \beta_1 t + \beta_2 x_{t,2} + \beta_3 x_{t,3} + \ldots + \beta_F x_{t,F} + \varepsilon_i,$$

for $t \in \{1, 2, \ldots, T\}$, and $x_{t,j} \in \{0, 1\}$ is an artificial variable such that $x_{t,j} = 1$ if $t \bmod F = j \bmod F$.

The *stationarity* is given by $(1 - p)$, where p is a p-value of the Augmented Dickey-Fuller Test of stationarity.

The last step is to set up appropriate thresholds of the support and confidence of the fuzzy GUHA method, and the reduction threshold. This is done using a grid optimization and cross-validation technique on the training set. In our case, we have used the 10-fold cross-validation, i.e., the training set was separated into ten subsets, with nine of them being used to generate linguistic descriptions using given parameter values, while the last (validation) subset being used to validate the forecasting performance. This was done ten times as the validation subset went through all ten subsets. The obtained forecasting performance measured by SMAPE was then aggregated by the arithmetic mean over all ten validations. This procedure was repeated as many times as we had combinations of parameters in our grid. Particularly, the grid was set up for confidence $\gamma \in \{0.5,\ 0.55,\ 0.6,\ 0.65,\ 0.7\}$, for the support $r \in \{0.025,\ 0.05,\ 0.075\}$, and for the reduction threshold $\varrho \in \{0.7, 0.8, 0.9, 0.95\}$.

The best performance on the cross-validation was achieved by the combination of $\gamma = 0.5$, $r = 0.05$, and $\varrho = 0.95$. We consider as best a such combination of parameters that achieves simultaneously best mean and standard deviation of SMAPE as follows: obtained SMAPE means are sorted and their rank is computed as well as ranks of SMAPE standard deviations. Then a combination with lowest sum of mean and standard deviation ranks is considered as the winner.

The winning combination was then used on the whole training set to generate rules that were then reduced by the size reduction algorithm with the chosen reduction threshold $\varrho = 0.95$. This way we have finally obtained the Fuzzy Rule-Based Ensemble whose performance can be independently evaluated and compared on the testing set, i.e., on a set that has not been previously used within any step of the training phase, so far.

4.6 Results

In order to judge its performance, the Fuzzy Rule-Based Ensemble was applied on the 1415 time series from the testing set, i.e., on all monthly, quarterly and yearly time series with odd IDs in the M3-Competition. Table 4.4 shows that the arithmetic mean and the standard deviation of SMAPE forecasting errors obtained on all testing time series is better for the Fuzzy Rule-Based Ensemble than any individual forecasting method from the R-package used in the ensemble (i.e., R-ARIMA, R-ES, R-THETA, R-RW). Moreover, the equal-weights, i.e., arithmetic mean (R-AM), and the three best methods from the M3-Competition according to the average precision on the testing set (M3-THETA, M3-ForecastPro, M3-ForcX) have been outperformed as well.

To evaluate the results even more thoroughly, we have also trained several state-of-the-art machine learning algorithms as well as some of the most well-known fuzzy inference methods that were at disposal to estimate weights for the ensemble from the time series features, as done by our R-FRBE method. Namely, the following non-fuzzy techniques were evaluated: Random Forest [78] (R-RF), Support Vector Machines with linear kernels [67] (R-SVM), Conditional Inference Random Forest (R-CForest) [59], CART (R-CART) of the *rpart* package [117], Neural Network (R-NNet) [119], Linear Regression Model (R-LM). Furthermore, several fuzzy inference mechanisms implemented in the *frbs* R-package [106] were used for the comparison too, namely: ANFIS (R-ANFIS) [66], DENFIS (R-DENFIS) [68], Hy-FIS (R-HyFIS) [71], WM model (R-WM) [124], Subtractive Clustering and Fuzzy C-Means Model (R-SBC) [127], and Genetic Fuzzy System for Fuzzy Rule Learning based on the MOGUL Methodology (R-GFS) [24].

The results of the comparison can be found in Table 4.4. To indicate superiority of our method, a statistical test of significance has been performed. Namely, we have performed the Wilcoxon signed rank test with continuity correction that assesses whether the forecasting accuracy (measured by SMAPE) mean rank differs for FRBE and any of the methods listed above. The null hypothesis (no difference) was rejected for all the methods except for the ensembles created with Random Forests (R-RF) and Support Vector Machines (R-SVM) where the difference was not considered statistically significant by the test. Also the robustness of the forecasts, i.e., the standard deviation of the SMAPE forecasting errors, was evaluated. For that purpose, multiple F-tests were performed to test the null hypothesis of ratio of variances being equal to 1.

All the statistically significant differences to the R-FRBE method are labelled with the star symbol ("*") in Table 4.4. The p-values of the statistical tests were

Method	SMAPE							
	Total		For $h = 6$		For $h = 8$		For $h = 18$	
	Avg	SD	Avg	SD	Avg	SD	Avg	SD
R-RF	13.14	14.03	15.79	14.45	8.97	10.74	14.16	14.88
R-SVM	13.26	14.19	15.94	14.59	8.95	10.79	14.33	15.07
R-FRBE	13.29	14.37	16.43	15.99	9.03	10.49	14.12	14.88
R-CForest	*13.30	14.10	16.00	14.48	9.09	10.76	14.30	14.99
R-CART	*13.33	14.14	16.11	14.60	9.04	10.79	14.34	14.99
R-Nnet	*13.37	14.11	16.01	14.62	9.15	10.76	14.41	14.94
R-LM	*13.38	14.11	15.99	14.39	9.10	10.76	14.46	15.05
R-SBC	*13.38	14.19	15.94	14.38	9.18	10.88	14.45	15.17
R-DENFIS	*13.45	14.31	16.12	14.71	9.17	10.83	14.52	15.25
R-HyFIS	*13.47	14.18	16.12	14.77	9.28	10.85	14.50	14.99
R-WM	*13.52	14.13	16.03	14.45	9.29	10.89	14.62	15.01
M3-THETA	*13.56	*15.42	17.66	18.73	8.89	10.62	14.18	15.29
R-GFS	*13.66	14.22	16.10	14.46	9.40	10.83	14.82	15.19
R-AM	*13.66	14.22	16.10	14.46	9.40	10.83	14.82	15.19
M3-ForecastPro	*13.67	*15.50	17.42	18.89	9.76	12.38	14.04	14.82
M3-ForcX	*13.76	14.94	16.71	16.30	9.35	11.88	14.76	15.26
R-ES	*13.95	15.23	17.34	17.27	9.67	11.36	14.69	15.54
R-ARIMA	*14.58	*16.77	17.87	18.12	9.80	12.71	15.62	17.48
R-THETA	*14.73	*15.33	17.34	16.40	10.44	11.40	15.83	16.18
R-ANFIS	*14.73	*15.33	17.34	16.39	10.44	11.40	15.83	16.18
R-RW	*16.53	*17.20	18.32	14.94	11.33	13.02	18.48	19.40

Table 4.4: Average and standard deviation of the SMAPE forecasting errors. Stars in the second and third column indicate statistically significant difference to the R-FRBE method. The tests were evaluated only for total results.

adjusted with the Benjamini-Hochberg correction that controls the false discovery rate. The significance level was set to $\alpha = 0.05$.

$$
\begin{aligned}
\mathcal{R}_1 &:= \text{IF } Skewness \text{ is } QR\,Sm \text{ THEN } w_{AR} \text{ is } QR\,Bi \\
\mathcal{R}_2 &:= \text{IF } SeasonStrength \text{ is } Ve\,Bi \text{ THEN } w_{AR} \text{ is } QR\,Bi \\
\mathcal{R}_3 &:= \text{IF } SeasonStrength \text{ is } QR\,Bi \text{ AND } Frequency \text{ is } Me \text{ THEN } w_{AR} \text{ is } Ex\,Bi \\
\mathcal{R}_4 &:= \text{IF } Lenght \text{ is } QR\,Me \text{ AND } SeasonStrength \text{ is } QR\,Sm \text{ AND} \\
&\qquad Kurtosis \text{ is } ML\,Me \text{ THEN } w_{AR} \text{ is } Ro\,Sm \\
&\quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots
\end{aligned}
\tag{4.6.1}
$$

Figure 4.2: An example of a part of the post-processed rule base for the R-ARIMA method.

In order to emphasize the linguistic nature of the approach, we present Figure 4.2 that provides readers with a part of one of the linguistic descriptions generated by the fuzzy GUHA method. The fuzzy rules symbolically displayed in Figure 4.2 can be easily read as conditional sentences of a natural language. Recall, that we have chosen the weight to be proportional to the expected method accuracy and thus the weight and the accuracy may be freely replaced. This makes the rule even more interpretable, which underlines the goal of our approach. Due to this nature of the model, one can easily extend the linguistic descriptions by new knowledge, no matter if obtained from an expert or again in a data-driven way.

Remark 7 *As one may see, most of the rules for R-ARIMA use the consequents derived from the linguistic expression* `Big`*, which is natural as R-ARIMA provided rather good results. For some case, specified, e.g., by the antecedent of the rule* $\mathcal{R}_3$*, the accuracy (and consequently the weight) was determined even as extremely good. However, the fuzzy GUHA also found a case, specified, e.g., by the antecedent of the rule* $\mathcal{R}_4$*, where the accuracy of the R-ARIMA method was found roughly small.*

The presented approach using the fuzzy GUHA and the linguistic descriptions is not restricted only to this particular Fuzzy Rule-Based Ensemble model. Users can build their own Fuzzy Rule-Based Ensemble (using the linguistic association mining) for specific time series (e.g. for daily time series from the finance domain) where they could use less variables, obtain less rules and get even more readable results, which might be again very helpful in further decision-making processes, not only in the prediction itself.

4.7 Conclusions

The introduced fuzzy rule-based ensemble has been "equipped" with fuzzy rule bases that were generated by the fuzzy GUHA method on 1414 time series from the training data set. The performance of the final model has been verified on the testing set composed of comparably very high number of 1415 times series, which makes this study undoubtedly comprehensive yet still possibly extendable.

The obtained results showed an improvement in the accuracy as well as in the standard deviation of the accuracy that confirms the improvement in the sense of "robustness".

Undoubtedly, the results confirm some sort of improvement. One could surely express objections to the too slight improvement and also to the too difficult and technologically demanding approach. Both objections have to be taken seriously as they have reasonable cores.

Related to the first objection, we have to stress that we have tested the improvement in accuracy not only compared to the arithmetic mean but also compared to all the individual methods (with p-value adjustment for multiple comparisons).

The suggested Fuzzy Rule-Based Ensemble method was found significantly better

in accuracy by the Wilcoxon test than any of the used individual method from the R package *forecast*. Any of the M3-methods was beaten as well. Moreover, the same can be stated about the equal weights method represented in this study by R-AM. The same can be stated about all the ensembles built in the same way as our Fuzzy Rule-Based Ensemble while using a different fuzzy method for the determination of the weights. We have used six methods from *frbs*, the key package for fuzzy methods in R, namely ANFIS, DENFIS, Hy-FIS, Wang-Mendel's generating fuzzy rules by learning from examples (R-WM), Yager-Filev's Subtractive Clustering and Fuzzy C-Means Model (R-SBC) [127], and the MOGUL Methodology (R-GFS). Apart from the ANFIS, all these fuzzy methods confirmed that they can be very useful as the ensembles based on them outperformed all individual methods from the *forecast* package and also the R-AM equal weights ensemble. However, the proposed Fuzzy Rule-Based Ensemble outperformed all of them which was confirmed by the above mentioned statistical significance test.

In order not to restrict our focus only on fuzzy approaches to determine the weights, also other very well-known machine learning methods were chosen. As one may see from Table 4.4, four out of six chosen methods were outperformed by the Fuzzy Rule-Based Ensemble and this fact was again confirmed by the statistical test. Two methods, namely the support vector machine and the random forrest, obtained higher accuracy. However, note, that this has not been confirmed by the chosen test as statistically significant!

This means that the results provide maybe slight yet statistically significant improvement. This is not that much surprising, having in mind the extremely high number of time series in the testing set. However, this is nothing against the validity of the results. Vice-versa, the bigger the testing set, the better for the experimental justification.

Regarding the second objection, let us stress that the difficulty appears only in the construction phase. In the application phase, a user only takes a rather simple tool built into an R-package *lfl* [18] that automatically determines a given time series features, uses the pre-determined fuzzy rules to set-up weights of individual methods, performs individual method forecasts, combines them accordingly to the determined weights, and finally, provides a user with a single accurate and robust forecast. Let us note that R-packages became very useful and standard statistical tools, and recently they turn into a preferable way of how to disseminate the state-of-art and the latest methods under the open source platform also in the fuzzy community, see [58, 105, 106].

Based on the actual investigation, we can summarize the following conclusions:

- using the standard forecasting methods implemented in the R-package `forecast` is not a bad choice even if compared to top commercial products;
- ensembling, even in the simple equal weights setting, is a valuable step that may be very helpful;
- in building a more sophisticated ensemble, fuzzy GUHA is very appropriate candidate that has been confirmed to provide better results than the equal weights, the chosen fuzzy methods or most of the chosen machine learning approaches;
- the advantage of the approach based on the fuzzy GUHA lies in the interpretable form that is easy to update by new knowledge, no matter if mined from data or obtained from an expert;
- the use of the concept of coverage of data by a linguistic description is highly

desirable, it can be helpful in setting the GUHA parameters and also in reduction algorithms;

- the size reduction algorithm proposed in this paper is characterized by an enormous reduction with keeping analogous behaviour and thus, it may be suggested also for other application areas of the association mining algorithms.

Regarding the interpretability issues, which is unquestionably an important point, let us note, that there is a difference compared to the interpretability of econometrical models, where the interpretability is on the level of the time series generating process. For example, in case of the decomposition model, the model itself gives nicely interpretable information about trends, seasonalities etc. The Fuzzy Rule-Based Ensemble model as suggested, does not give any information about the time series generating process itself. However, it provides us with information, which forecasting models are of which accuracy expectations, which is undoubtedly also information of a significant importance.

Bibliography

[1] *Data and metadata reporting and presentation handbook*, OECD, 2005.

[2] R. Agrawal and R. Srikant, *Fast algorithms for mining association rules*, Proc. 20th Int. Conf. on Very Large Databases (Chile), AAAI Press, pp. 487–499.

[3] R. R. Andrawis and A. F. Atiya, *A new Bayesian formulation for Holt's exponential smoothing*, Journal of Forecasting **28** (2009), no. 3, 218–234.

[4] J. S. Armstrong, *Long-range forecasting*, Wiley New York ETC., 1985.

[5] ———, *Evaluating methods*, Principles of Forecasting: A handbook for reasearchers and practitioners (J. S. Armstrong, ed.), Kluwer Academic Publishers, Boston/Dordrecht/London, 2001, pp. 443–473.

[6] J. S. Armstrong, M. Adya, and F. Collopy, *Rule-Based Forecasting Using Judgment in Time Series Extrapolation*, Principles of Forecasting: A handbook for reasearchers and practitioners (J. S. Armstrong, ed.), Kluwer Academic Publishers, Boston/Dordrecht/London, 2001.

[7] J. S. Armstrong and F. Collopy, *Error measures for generalizing about forecasting methods: Empirical comparisons*, International Journal of Forecasting **8** (1992), 69–80.

[8] J. Aznarte, J. Benítez, and J. Castro, *Smooth transition autoregressive models and fuzzy rule-based systems: Functional equivalence and consequences*, Fuzzy Sets and Systems **158** (2007), 2734–2745.

[9] R. Babuška and M. Setnes, *Data-driven construction of transparent fuzzy models*, Fuzzy Algorithms for Control (H.B. Verbruggen, H.-J. Zimmermann, and R. Babuška, eds.), Kluwer Academic Publishers, Boston, 1999, pp. 83–106.

[10] J. M. Bates and C. W. J. Granger, *Combination of forecasts*, Operational Research Quarterly **20** (1969), 451–468.

[11] R. K. Belew, J. McInerney, and N. Schraudolph, *Evolving networks: Using the genetic algorithm with connectionist learning*, Proc. 2nd Workshop on Artificial Life, pp. 511–547.

[12] U. Bodenhofer and P. Bauer, *Interpretability of linguistic variables: a formal account*, Kybernetika **2** (2005), 227–248.

[13] A. Bovas and J. Ledolter, *Statistical methods for forecasting*, Wiley, New York, 2003.

[14] G. Box and G. Jenkins, *Time series analysis: Forecasting and control*, Holden-Day, San Francisco, 1976.

[15] P. Brockwell and R. Davis, *Time series: Theory and methods (springer series in statistics), 2nd ed.*, Springer-Verlag, Heidelberg, 1998.

[16] R. G. Brown, *Statistical forecasting for inventory control*, McGraw-Hill, New York, 1959.

[17] M. Burda, *Interest measures for fuzzy association rules based on expectations of independence*, Advances in Fuzzy Systems **2014** (2014), 1–7.

[18] ———, *lfl: Linguistic fuzzy logic (r package on cran)*, 2015.

[19] M. Burda and M. Štěpnička, *Reduction of fuzzy rule bases driven by the coverage of training data*, Proc. 16th World Congress of the International Fuzzy Systems Association and 9th Conference of the European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT 2015) (Gijón), Advances in Intelligent Systems Research, Atlantic Press, 2015.

[20] R. Bělohlávek and V. Novák, *Learning rule base in linguistic expert systems*, Soft Computing **7** (2002), 79–88.

[21] J. Casillas, O. Cordón, F. Herrera Triguero, and L. Magdalena (eds.), *Interpretability issues in fuzzy modeling (studies in fuzziness and soft computing vol. 128)*, Springer, Heidelberg, 2003.

[22] V. Cherkassy and Y. Ma, *Practical Selection of SVM Parameters and Noise Estimation for SVM Regression*, Neural Networks **17** (2004), no. 1, 113–126.

[23] F. Collopy and J. S. Armstrong, *Rule-Based Forecasting: Development and Validation of an Expert Systems Approach to Combining Time Series Extrapolations*, Management Science **38** (1992), 1394–1414.

[24] O. Cordón, M. J. del Jesus, F. Herrera, and M. Lozano, *Mogul: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach*, International Journal of Intelligent Systems **14** (1998), 1123–1153.

[25] P. Cortez, *Data Mining with Neural Networks and Support Vector Machines using the R/rminer Tool*, Proc. 10th Industrial Conference on Advances in Data Mining: Applications and Theoretical Aspects, 2010, pp. 572–583.

[26] P. Cortez, M. Rio, M. Rocha, and P. Sousa, *Internet Traffic Forecasting using Neural Networks*, Proc. 2006 International Joint Conference on Neural Networks (IJCNN 2006) (Vancouver, Canada), IEEE, 2006, pp. 4942–4949.

[27] P. Cortez, M. Rocha, and J. Neves, *Evolving Time Series Forecasting ARMA Models*, Journal of Heuristics **10** (2004), no. 4, 415–429.

[28] ———, *Time Series Forecasting by Evolutionary Neural Networks*, ch. III, pp. 47–70, Idea Group Publishing, USA, 2006.

[29] S. Crone and N. Kourentzes, *Feature selection for time series prediction - A combined filter and wrapper approach for neural networks*, Neurocomputing **73** (2010), 1923–1936.

[30] S. F. Crone, M. Hibon, and K. Nikolopoulos, *Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction*, International Journal of Forecasting **27** (2011), no. 3, 635–660.

[31] S. F. Crone and N. Kourentzes, *Naive Support Vector Regression and Multilayer Perceptron Benchmarks for the 2010 Neural Network Grand Competition (NNGC) on Time Series Prediction*, Proc. 2010 IEEE Int. Joint Conf.on Neural Networks (IJCNN 2010), IEEE (Barcelona, Spain), 2010, pp. 2878–2885.

[32] G. Cybenko, *Approximation by superposition of a sigmoidal function*, Mathematics of Control, Signals and Systems (1989), 303–314.

[33] M. De Cock and E.E. Kerre, *Fuzzy modifiers based on fuzzy relations*, Information Sciences **160** (2004), 173–199.

[34] D. Dubois and H. Prade, *What are fuzzy rules and how to use them*, Fuzzy Sets and Systems **84** (1996), 169–185.

[35] A. Dvořák, H. Habiballa, V. Novák, and V. Pavliska, *The software package LFLC 2000 - its specificity, recent and perspective applications*, Computers in Industry **51** (2003), 269–280.

[36] T. Edwards, D. S. W. Tansley, R. J. Frank, N. Davey, and Northern Telecom (nortel Limited, *Traffic trends analysis using neural networks*, Proceedings of the International Workshop on Applications of Neural Networks to Telecommuncations, 1997, pp. 157–164.

[37] A. P. Engelbrecht, *Computational intelligence: an introduction*, Wiley, 2007.

[38] D. B. Fogel, *Evolutionary computation: Toward a new philosophy of machine intelligence*, third ed., IEEE Press Series on Computational Intelligence, Wiley-IEEE Press, December 2005.

[39] R. J. Frank, N. Davey, and S. P. Hunt, *Time series prediction and neural networks*, J. Intell. Robotics Syst. **31** (2001), 91–103.

[40] M. Frean, *The upstart algorithm: a method for constructing and training feed-forward neural networks*, Neural Computation **2** (1990), no. 2, 198–209.

[41] S. Galichet and L. Foulloy, *Size reduction in fuzzy rulebases*, Proc. IEEE International Conference On Systems, Man and Cybernetics (San Diego), 1998, pp. 2107–2112.

[42] L. Geng and H. J. Hamilton, *Interestingness measures for data mining: A survey*, ACM Computing Surveys **38** (2006), no. 3.

[43] D. H. Glass, *Entailment and symmetry in confirmation measures of interestingness*, Information Sciences **279** (2014), 552 – 559.

[44] David H. Glass, *Fuzzy confirmation measures*, Fuzzy Sets Systems **159** (2008), no. 4, 475–490.

[45] David H. Glass, *Confirmation measures of association rule interestingness*, Knowledge-Based Systems **44** (2013), 65 – 77.

[46] L. Glass and M. Mackey, *Oscillation and chaos in physiological control systems*, Science **197** (1977), 287–289.

[47] R. Goodrich, *The forecast pro methodology*, International Journal of Forecasting **16** (2000), no. 4, 533–535.

[48] P. Hájek, *The question of a general concept of the GUHA method*, Kybernetika **4** (1968), 505–515.

[49] P. Hájek, *Metamathematics of fuzzy logic*, Trends in Logic, vol. 4, Kluwer Academic Publishers, Dordrecht, 1998.

[50] P. Hájek and T. Havránek, *Mechanizing hypothesis formation: Mathematical foundations for a general theory*, Springer-Verlag, Berlin/Heidelberg/New York, 1978.

[51] P. Hájek, M. Holeňa, and J. Rauch, *The GUHA method and its meaning for data mining*, Journal of Computer and Systems Sciences **76** (2010), 34–48.

[52] E. Hajičová, B. Partee, and P. Sgall, *Topic-focus articulation, tripartite structures, and semantic content*, Kluwer Academic Publishers, Dordrecht, 1998.

[53] Maria Halkidi and Michalis Vazirgiannis, *Quality assessment approaches in data mining*, The Data Mining and Knowledge Discovery Handbook, 2005, pp. 661–696.

[54] J. D. Hamilton, *Time series analysis*, Princeton University Press, New Jersey, 1994.

[55] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*, 2nd ed., Springer-Verlag, NY, USA, 2008.

[56] W. He, Z. Wang, and H. Jiang, *Model optimizing and feature selecting for support vector regression in time series forecasting*, Neurocomputing **72** (2008), no. 1-3, 600–611.

[57] C. C. Holt, *Forecasting trends and seasonals by exponentially weighted averages*, O.N.R. Memorandum **52** (1957).

[58] Kurt Hornik and David Meyer, *Generalized and customizable sets in R*, Journal of Statistical Software **31** (2009), 1–27.

[59] Torsten Hothorn, Peter Buehlmann, Sandrine Dudoit, Annette Molinaro, and Mark Van Der Laan, *Survival ensembles*, Biostatistics **7** (2006), no. 3, 355–373.

[60] K. Huarng, *Heuristic models of fuzzy time series for forecasting*, Fuzzy Sets and Systems **123** (2001), 369–386.

[61] R. Hyndman and A. Koehler, *Another look at measures of forecast accuracy*, International Journal of Forecasting **22** (2006), 679–688.

[62] R. J. Hyndman, *Time series data library*, http://robjhyndman.com/TSDL/.

[63] R. J. Hyndman and G. Athanasopoulos, *Forecasting principles and practice*, OTexts, 2013.

[64] Rob J Hyndman, George Athanasopoulos, Slava Razbash, Drew Schmidt, Zhenyu Zhou, Yousaf Khan, and Christoph Bergmeir, *forecast: Forecasting functions for time series and linear models*, 2013, R package version 4.06.

[65] R. A. Jacobs, *Increased rates of convergence through learning rate adaptation*, Neural Networks **1** (1988), no. 4, 295–307.

[66] J.-S. R. Jang, *Anfis: Adaptive-network-based fuzzy inference system*, IEEE Transactions on Systems, Man, and Cybernetics **23** (1993), 665–685.

[67] Alexandros Karatzoglou, Alex Smola, Kurt Hornik, and Achim Zeileis, *kernlab – an S4 package for kernel methods in R*, Journal of Statistical Software **11** (2004), no. 9, 1–20.

[68] N. Kasabov and Q. Song, *Denfis: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction*, IEEE Transactions on Fuzzy Systems **10** (2002), 144–154.

[69] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, *Dimensionality reduction for fast similarity search in large time series databases*, Knowledge and Information Systems **3** (2001), 263–286.

[70] R. Kewley, M. Embrechts, and C. Breneman, *Data Strip Mining for the Virtual Design of Pharmaceuticals with Neural Networks*, IEEE Trans Neural Networks **11** (2000), no. 3, 668–679.

[71] Jaesoo Kim and Nikola K. Kasabov, *Hyfis: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems*, Neural Networks **12** (1999), no. 9, 1301–1319.

[72] E. P. Klement, R. Mesiar, and E. Pap, *Triangular norms*, Trends in Logic, vol. 8, Kluwer Academic Publishers, Dordrecht, 2000.

[73] J. Kupka and I. Tomanová, *Some extensions of mining of linguistic associations*, Neural Network World **20** (2010), 27–44.

[74] ———, *Dependencies among attributes given by fuzzy confirmation measures*, Expert Systems with Applications **39** (2012), no. 9, 7591–7599.

[75] C. Lemke and B. Gabrys, *Meta-learning for time series forecasting in the nn gc1 competition*, Proc. 16th IEEE Int. Conf. on Fuzzy Systems (Barcelona), 2010, pp. 2258–2262.

[76] G. Leng, T. McGinnity, and G. Prasad, *An approach for on-line extraction of fuzzy rules using a self-organising fuzzy neural network*, Fuzzy Sets and Systems **150** (2005), 211–243.

[77] M.-T. Li, S.-C. Kuo, Y.-C. Cheng, and C. C. Chen, *Deterministic vector long-term forecasting for fuzzy time series*, Fuzzy Sets and Systems **161** (2010), 1852–1870.

[78] Andy Liaw and Matthew Wiener, *Classification and regression by randomforest*, R News **2** (2002), no. 3, 18–22.

[79] E. Lughofer and E. Hüllermeier, *On-line redundancy elimination in evolving fuzzy regression models using a fuzzy inclusion measure*, Proc. of EUSFLAT-LFA 2011 (Aix-les-Bains), July 2011, pp. 380–387.

[80] S. Makridakis, R. Fildes, M. Hibon, R. Lewandowski, J. Newton, E. Parzen, and R. Winkler, *The accuracy of extrapolation (time-series) methods - results of a forecasting competition*, Journal of Forecasting **1** (1982), 111–153.

[81] S. Makridakis and M. Hibon, *The M3–Competition: results, conclusions and implications*, International Journal of Forecasting **16** (2000), 451–476.

[82] S. Makridakis, S. Wheelwright, and R. Hyndman, *Forecasting: methods and applications*, John Wiley & Sons, USA, 2008.

[83] E. H. Mamdani and S. Assilian, *An experiment in linguistic synthesis with a fuzzy logic controller*, Int. J. Man-Mach. Stud. **7** (1975), 1–13.

[84] K. Miiller, A. Smola, G. Ratsch, B. Scholkopf, J. Kohlmorgen, and V. Vapnik, *Predicting time series with support vector machines*, Proc. 7th International Conference on Artificial Neural Networks, Springer, 1997, pp. 999–1004.

[85] G. F. Miller, P. M. Todd, S. U. Hegde, and U. Shailesh, *Designing neural networks using genetic algorithms*, Proc. 3rd International Conference on Genetic Algorithms, 1989, pp. 379–384.

[86] P. Newbold and C. W. J. Granger, *Experience with forecasting univariate time series and combination of forecasts*, Journal of the Royal Statistical Society Series a-Statistics in Society **137** (1974), 131–165.

[87] V. Novák, *Základy fuzzy modelování*, Technická literatura BEN, Praha, in Czech, 2000.

[88] ———, *Perception-based logical deduction*, Computational Intelligence, Theory and Applications (Berlin) (B. Reusch, ed.), Advances in Soft Computing, Springer, 2005, pp. 237–250.

[89] ———, *A comprehensive theory of trichotomous evaluative linguistic expressions*, Fuzzy Sets and Systems **159** (2008), no. 22, 2939–2969.

[90] V. Novák and A. Dvořák, *Formalization of commonsense reasoning in fuzzy logic in broader sense*, Applied and Computational Mathematics **10** (2011), 106–121.

[91] V. Novák and S. Lehmke, *Logical structure of fuzzy IF-THEN rules*, Fuzzy Sets and Systems **157** (2006), no. 15, 2003–2029.

[92] V. Novák and I. Perfilieva, *Evaluating linguistic expressions and functional fuzzy theories in fuzzy logic*, Computing with Words in Information/Intelligent Systems 1 (L.A. Zadeh and J. Kacprzyk, eds.), Springer-Verlag, Heidelberg, 1999, pp. 383–406.

[93] V. Novák, I. Perfilieva, and A. Dvořák, *Insight into fuzzy modeling*, John Wiley & Sons, USA, 2015.

[94] V. Novák, I. Perfilieva, A. Dvořák, Q. Chen, Q. Wei, and P. Yan, *Mining pure linguistic associations from numerical data*, International Journal of Approximate Reasoning **48** (2008), 4–22.

[95] V. Novák, I. Perfilieva, and J. Močkoř, *Mathematical principles of fuzzy logic*, Kluwer Academic Publishers, Boston, 1999.

[96] V. Novák, M. Štěpnička, A. Dvořák, I. Perfilieva, V. Pavliska, and L. Vavříčková, *Analysis of seasonal time series using fuzzy approach*, International Journal of General Systems **39** (2010), 305–328.

[97] I. Nunn and T. White, *The application of antigenic search techniques to time series forecasting*, GECCO, 2005, pp. 353–360.

[98] A. K. Palit and D. Popovic, *Computational intelligence in time series forecasting: Theory and engineering applications (advances in industrial control)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[99] J. Peralta, X. Li, G. Gutierrez, and A. Sanchis, *Time series forecasting by evolving artificial neural networks using genetic algorithms and differential evolution*, IJCNN, 2010.

[100] I. Perfilieva, *Fuzzy transforms: theory and applications*, Fuzzy Sets and Systems **157** (2006), 993–1023.

[101] I. Perfilieva, V. Novák, V. Pavliska, A. Dvořák, and M. Štěpnička, *Analysis and prediction of time series using fuzzy transform*, Proc. IEEE World Congress on Computational Intelligence, 2008, pp. 3875–3879.

[102] I. Perfilieva and R. Valášek, *Fuzzy transforms in removing noise*, Computational Intelligence, Theory and Applications (Berlin) (B. Reusch, ed.), Advances in Soft Computing, Springer, 2005, pp. 221–230.

[103] F. M. Pouzols, A. Lendasse, and A. Barriga Barros, *Autoregressive time series prediction by means of fuzzy inference systems using nonparametric residual variance estimation*, Fuzzy Sets and Systems **161** (2010), 471–497.

[104] S. Price, *Mining the past to determine the future: Comments*, International Journal of Forecasting **25** (2009), no. 3, 452–455.

[105] Lala Septem Riza, Christoph Bergmeir, Francisco Herrera, and Jose Manuel Benitez, *Learning from data using the R package "frbs"*, Proc. IEEE International Conference On Fuzzy Systems (Beijing), 2014, pp. 2149–2155.

[106] ———, *frbs: Fuzzy rule-based systems for classification and regression in R*, Journal of Statistical Software (to appear).

[107] H. J. Rong, N. Sundararajan, G. B. Huang, and P. Saratchandran, *Sequential adaptive fuzzy inference system (safis) for nonlinear system identification and prediction*, Fuzzy Sets and Systems **157** (2006), 1260–1275.

[108] Emilia Sainio, Esko Turunen, and Radko Mesiar, *A characterization of fuzzy implications generated by generalized quantifiers*, Fuzzy Sets and Systems **159** (2008), no. 4, 491 – 499.

[109] M. Setnes, *Fuzzy rule base simplification using similarity measures*, Ph.D. thesis, MSc Thesis. Delft University of Technology, Defit, Netherlands, 1995.

[110] M. Setnes, V. Lacrose, and A. Titli, *Complexity reduction methods for fuzzy systems*, Fuzzy Algorithms for Control (H.B. Verbruggen, H.-J. Zimmermann,

and R. Babuška, eds.), Kluwer Academic Publishers, Boston, 1999, pp. 185–218.

[111] P. Sgall, E. Hajičová, J. Panevová, and J. Mey, *The meaning of the sentence in its semantic and pragmatic aspects*, Kluwer Academic Publishers, Boston, 1986.

[112] D. Sikora, M. Štěpnička, and L. Vavříčková, *Fuzzy rule-based ensemble forecasting: Introductory study*, Synergies of Soft Computing and Statistics for Intelligent Data Analysis, Advances in Intelligent Systems and Computing, vol. 190, Springer-Verlag, 2013, pp. 379–387.

[113] ———, *On the potential of fuzzy rule-based ensemble forecasting*, International Joint Conference CISIS'12 - ICEUTE'12 - SOCO'12 SPECIAL SESSIONS, Advances in Intelligent Systems and Computing, vol. 189, Springer-Verlag, 2013, pp. 487–496.

[114] A. Smola and B. Schölkopf, *A tutorial on support vector regression*, Statistics and Computing **14** (2004), 199–222.

[115] Q. Song and B. Chissom, *Fuzzy time series and its models*, Fuzzy Sets and Systems **54** (1993), 269–277.

[116] T. Takagi and M. Sugeno, *Fuzzy identification of systems and its applications to modeling and control*, IEEE Transactions on Systems, Man and Cybernetics **15** (1985), 116–132.

[117] Terry Therneau, Beth Atkinson, and Brian Ripley, *rpart: Recursive partitioning and regression trees*, 2015, R package version 4.1-9.

[118] F. M. Tseng, G. H. Tzeng, Yu, and B. J. C. Yuan, *Fuzzy ARIMA model for forecasting the foreign exchange market*, Fuzzy Sets and Systems **118** (2001), 9–19.

[119] W. N. Venables and B. D. Ripley, *Modern applied statistics with s*, fourth ed., Springer, New York, 2002, ISBN 0-387-95457-0.

[120] M. Štěpnička, U. Bodenhofer, M. Daňková, and V. Novák, *Continuity issues of the implicational interpretation of fuzzy rules*, Fuzzy Sets and Systems **161** (2010), 1959–1972.

[121] M. Štěpnička and B. De Baets, *Implication-based models of monotone fuzzy rule bases*, Fuzzy Sets and Systems (in press).

[122] M. Štěpnička, A. Dvořák, V. Pavliska, and L. Vavříčková, *A linguistic approach to time series modeling with the help of the f-transform*, Fuzzy sets and systems **180** (2011), 164–184.

[123] M. Štěpnička and O. Polakovič, *A neural network approach to the fuzzy transform*, Fuzzy sets and Systems **160** (2009), 1037–1047.

[124] L.-X. Wang and J.M. Mendel, *Generating fuzzy rules by learning from examples*, Systems, Man and Cybernetics, IEEE Transactions on **22** (1992), no. 6, 1414–1427.

[125] W. Wang, Z. Xu, W. Lu, and X. Zhang, *Determination of the spread parameter in the Gaussian kernel for classification and regression*, Neurocomputing **55** (2003), no. 3, 643–663.

[126] D. Whitley and T. Hanson, *Optimizing neural networks using faster, more accurate genetic search*, Proc. 3rd International Conference on Genetic Algorithms, 1989, pp. 391–396.

[127] R. R. Yager and D. P. Filev, *Generation of fuzzy rules by mountain clustering*, Journal of Intelligent and Fuzzy Systems **2** (1994), 209–219.

[128] L. A. Zadeh, *Fuzzy sets*, Inf. Control **8** (1965), 338–353.

[129] L. A. Zadeh, *Outline of a new approach to the analysis of complex systems and decision processes*, IEEE Trans. on Systems, Man, and Cybernetics **3** (1973), no. 1, 28–44.

[130] L. A. Zadeh, *The concept of a linguistic variable and its application to approximate reasoning I*, Inform. Sci. **8** (1975), 199–250.

[131] ———, *The concept of a linguistic variable and its application to approximate reasoning I–III*, Inform. Sci. 8 (1975) 199–250, 8 (1975) 301–357, 9 (1975) 43–80, 1975.

[132] ———, *The concept of a linguistic variable and its application to approximate reasoning II*, Inform. Sci. **8** (1975), 301–357.

[133] ———, *The concept of a linguistic variable and its application to approximate reasoning III*, Inform. Sci. **9** (1975), 43–80.

[134] ———, *Precisiated natural language*, AI Magazine **25** (2004), 74–91.

[135] A. Zell, G. Mamier, R. Hübner, N. Schmalzl, T. Sommer, and M. Vogt, *SNNS: An efficient simulator for neural nets*, MASCOTS '93: Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems (San Diego, CA, USA), Society for Computer Simulation International, 1993, pp. 343–346.