

Linguistic Descriptions As a Modeling Tool For Multivariate Time Series

Pavel Rusnok¹

¹Centre of Excellence IT4Innovations, Division of the University of Ostrava
Institute for Research and Applications of Fuzzy Modeling
30. dubna 22, Ostrava, Czech Republic

Abstract

We propose linguistic associations mining as a technique to create the models of the multivariate time series. We define various linguistic evaluative expressions on the range of the values of the time series and variables derived from them. We mine linguistic associations then and interpret them as IF-THEN rules in the framework of Perception based Logic Deduction (PbLD). The mined rules provide the linguistic descriptions of various relationships between the time series. We showcase our suggested methodology in a macroeconomic example where we compare our approach with Dynamic Stochastic General Equilibrium (DSGE) model, that is frequently used in the macroeconomic modeling.

Keywords: multivariate time series, evaluative linguistic expressions, association analysis, Perception based Logic Deduction

1. Introduction

Time series analysis is a vast field of a research with many well established mathematical models; the ARIMA models, exponential smoothing and decomposition being the prominent examples, see e. g., [1, 2]. In the recent years the interest in time series modeling is even increasing as the various smoothing, dimension reduction and representation methods are suggested, see [3] and references in it. Also many attempts to use computational intelligence were suggested, see [4] and the references given there. Our work belongs to the latter.

This work attempts to show the usefulness of linguistic expressions as a descriptive modeling tool for the multivariate time series. They were successfully studied in an application to the univariate time series [5], where also F-transform (see [6]) was used as a smoothing and dimension reduction technique. We restrict ourselves to using only the linguistic expressions as a modeling tool and move from univariate times series to multivariate times series.

The basic idea is to extend data containing multivariate time series with other descriptive variables derived from them and use a data mining method to extract the relationships between the various time series and their features. We mine association rules with exactly one consequent (modeled time series),

which we interpret as IF-THEN rules, that are then used in an implication-based inference. After defuzzification we obtain the values of the modeled time series.

An example of an IF-THEN rule is:

IF X_1 is *Small* **AND** X_2 is *Big*
THEN X_3 is *Medium*,

where X_3 is a modeled time series and X_1, X_2 might be some derived variables from X_3 or other time series. Our method is applicable to an arbitrary number of the time series, even to an univariate case.

The next section provides all the necessary theoretical background for explaining our model. In Section 3 we describe our modeling approach with an example based on european macroeconomic data and we conclude.

2. Preliminaries

Please recall the following basic notions that are used in the article. A t-norm is a function $\otimes : [0, 1] \times [0, 1] \rightarrow [0, 1]$ that is associative, commutative, monotone and 1 acts as an identity element. An example of a t-norm is a minimum function that we denote as \wedge . A negation is a function $\neg : [0, 1] \rightarrow [0, 1]$ such that $\neg x = 1 - x$. The last connective we use is the Łukasiewicz implication $x \rightarrow y = \min\{1, 1 - x + y\}$

2.1. Evaluative linguistic predications

We introduce basics of evaluative linguistic expressions theory (for short *evaluative expressions*). For more detailed study see [7]. They are expressions of natural language used in decision making processes. Typical examples are *significantly different*, *very small*, etc. The possibility to model them mathematically was shown by L. A. Zadeh in [8, 9].

Evaluative expressions have the following syntax:

⟨linguistic hedge⟩⟨atomic evaluative expression⟩,

where *atomic evaluative expression* is one of the *canonical* adjectives *small*, *medium*, *big*. Obviously these words are usually replaced by other words as “thin”, “thick”, “old”, “new” etc., depending on the context, and complemented by a middle term, such as “medium”, “average”, etc.

Hedge	Abbreviation
extremely	Ex
significantly	Si
very	Ve

Table 1: Linguistic hedges with narrowing effect

Hedge	Abbreviation
rather	Ra
more or less	ML
roughly	Ro
quite roughly	QR
very roughly	VR
very very roughly	VV

Table 2: Linguistic hedges with widening effect

Linguistic hedges are adverbs that make the meaning of evaluative expressions more precise. They have either *narrowing* or *widening* effect. See Tables 1 and 2 for the examples of such adverbs. In the following, the evaluative expressions will be denoted as script letters \mathcal{A} , \mathcal{B} , etc.

The evaluative expressions are used to describe the values of some variable X . The resulting expression is called *evaluative linguistic predication* (in short *evaluative predication*) and have a form:

$$X \text{ is } \mathcal{A}. \quad (1)$$

Examples of evaluative predications are “price is very low”, “the difference of time series X_1 in time t is more or less big”, etc. To model the meaning of evaluative predications we have to distinguish between *intension* and *extension* of an evaluative predication in various *contexts*. We model the context as a triple of real numbers $w = \langle v_L, v_M, v_R \rangle$, where $[v_L, v_M]$ (resp. $[v_M, v_R]$) is a range of small (resp. big) numbers and v_M is the most typical medium number. We define the set of possible contexts as:

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

Then the *intension* of an evaluative predication “ X is \mathcal{A} ” is defined as a function

$$\text{Int}(X \text{ is } \mathcal{A}) : W \mapsto \mathcal{F}(\mathbb{R}),$$

which assigns to every context its extension (fuzzy set) and we denote the extension as:

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

where \subseteq denotes the relation of fuzzy subsethood. The model of evaluative predications is very general. To make it more concrete we define functions:

$$\begin{aligned} L_w(x) &= \frac{v_M - x}{v_M - v_L} \text{ for } x \in [v_L, v_M], \text{ else } L_w(x) = 0 \\ R_w(x) &= \frac{x - v_M}{v_R - v_M} \text{ for } x \in [v_M, v_R], \text{ else } R_w(x) = 0 \\ M_w(x) &= \neg L_w(x) \wedge \neg R_w(x), \end{aligned}$$

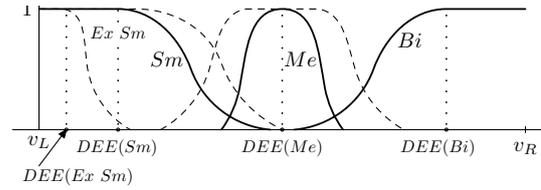


Figure 1: Example of fuzzy sets (extensions) and their defuzzification values.

where L_w , M_w and R_w model atomic evaluative expressions *Small*, *Medium* and *Big* respectively. Further we choose to model the linguistic hedges with a family of functions from $[0, 1]$ to $[0, 1]$ prescribed by the following formula:

$$\nu_{a,b,c}(x) = \begin{cases} 1, & c \leq x \\ 1 - \frac{(c-x)^2}{(c-b)(c-a)}, & b \leq x < c \\ \frac{(x-a)^2}{(b-a)(c-a)}, & a \leq x < b \\ 0, & x < a \end{cases}$$

We use also ⟨empty⟩ hedge and assign it one of the $\nu_{a,b,c}$ functions and thus all our extensions are non-linear. An example of one model is in Figure 1.

2.2. Linguistic IF-THEN rules, linguistic description

The evaluative predications are used in conditional statements as:

$$\mathcal{R} := \text{IF } X \text{ is } \mathcal{A} \text{ THEN } Y \text{ is } \mathcal{B}, \quad (2)$$

where \mathcal{A} , \mathcal{B} are evaluative expressions. The evaluative predication “ X is \mathcal{A} ” is called *antecedent* and the evaluative predication “ Y is \mathcal{B} ” is called *consequent*. We call conditional clauses in the form of (2) *fuzzy/linguistic IF-THEN rules* or just simply *IF-THEN rules*.

The set of fuzzy/linguistic IF-THEN rules is called *linguistic description* and is denoted as $LD = \{\mathcal{R}_1, \dots, \mathcal{R}_m\}$, where

$$\begin{aligned} \mathcal{R}_1 &:= \text{IF } X \text{ is } \mathcal{A}_1 \text{ THEN } Y \text{ is } \mathcal{B}_1 \\ &\vdots \\ \mathcal{R}_m &:= \text{IF } X \text{ is } \mathcal{A}_m \text{ THEN } Y \text{ is } \mathcal{B}_m \end{aligned} \quad (3)$$

The linguistic description is in fact a form of a structured text, that can be viewed as a model of specific behavior of a system in concern. In our case we will develop such sets of linguistic descriptions to model multivariate time series.

The *intension* of a fuzzy/linguistic IF-THEN rule \mathcal{R} in (2) is a function

$$\text{Int}(\mathcal{R}) : W \times W \rightarrow \mathcal{F}(\mathbb{R} \times \mathbb{R}), \quad (4)$$

which assigns to each context $w \in W$ of variable X and each context $w' \in W$ of variable Y a *fuzzy relation* on $w \times w'$, which is *extension* of (2).

We also distinguish between the sets of Intensions of antecedents and consequents of our fuzzy/linguistic IF-THEN rules.

$$\begin{aligned} Topic_{LD} &= \{\text{Int}(X \text{ is } \mathcal{A}_j) \mid j = 1, \dots, m\}, \\ Focus_{LD} &= \{\text{Int}(Y \text{ is } \mathcal{B}_j) \mid j = 1, \dots, m\}. \end{aligned}$$

Its denotation and usage in the inference is inspired by a linguistic phenomenon of topic-focus articulation. For more details see [10].

2.3. Ordering of linguistic predications

Perception-based logical deduction described in Section 2.4 filters out important part of linguistic description based on input value as a part of its inference mechanism. The whole process needs special ordering we will now define. First, we define ordering of linguistic hedges:

$$\text{Ex} \leq_H \text{Si} \leq_H \langle \text{empty} \rangle \leq_H \text{ML} \leq_H \text{Ro}.$$

The remaining hedges in Tables 1 and 2 are already written in an ordered way according to the ordering \leq_H .

Based on \leq_H we define an ordering \leq_{LE} as:

$$\langle \text{hedge}_i \rangle \mathcal{A} \leq_{LE} \langle \text{hedge}_j \rangle \mathcal{A},$$

where $\mathcal{A} \in \{\text{Sm}, \text{Me}, \text{Bi}\}$ and $\langle \text{hedge}_i \rangle \leq_H \langle \text{hedge}_j \rangle$.

To simplify notation we use \mathcal{A}_i instead of $\langle \text{hedge}_i \rangle \mathcal{A}$. Note that $\langle \text{hedge} \rangle \text{Sm}$ and $\langle \text{hedge} \rangle \text{Bi}$ are incomparable. Now we have everything to define ordering of the linguistic predications as:

$$(X \text{ is } \mathcal{A}_i) \preceq (X \text{ is } \mathcal{A}_j)$$

1. if

$$a_i < a_j$$

where

$$a_i := \neg(\text{Int}(X \text{ is } \mathcal{A}_i)(w))(u_0) \quad (5)$$

$$a_j := \neg(\text{Int}(X \text{ is } \mathcal{A}_j)(w))(u_0) \quad (6)$$

2. else if

$$a_i = a_j$$

and $\mathcal{A}_i \leq_{LE} \mathcal{A}_j$, where a_i, a_j are given by formulas (5) and (6), which are negated membership degrees of particular extensions at point u_0

For compound antecedents of the following form:

$$(X \text{ is } \mathcal{A}_i) = X_1 \text{ is } \mathcal{A}_{i_1} \text{ AND } \dots \text{ AND } X_K \text{ is } \mathcal{A}_{i_K},$$

$$(X \text{ is } \mathcal{A}_j) = X_1 \text{ is } \mathcal{A}_{j_1} \text{ AND } \dots \text{ AND } X_K \text{ is } \mathcal{A}_{j_K},$$

is the ordering extended component-wise:

$$\mathcal{A}_i \leq_{LE} \mathcal{A}_j \quad \text{if} \quad \mathcal{A}_{i_k} \leq_{LE} \mathcal{A}_{j_k} \quad \forall k = 1, \dots, K,$$

and an extension of the compound antecedents is given as a conjunction of its parts:

$$\begin{aligned} &(\text{Int}(X \text{ is } \mathcal{A}_i)(w_1, \dots, w_K))(u_1, \dots, u_K) = \\ &= \bigwedge_{k=1}^K (\text{Int}(X_k \text{ is } \mathcal{A}_{i_k})(w_k))(u_k). \end{aligned} \quad (7)$$

2.4. Perception-based Logical Deduction (PbLD)

We describe briefly PbLD in this section. For more detailed description see [11, 12].

We define a subset of all antecedents in a linguistic description LD called *local perception*

$$LPerc_{LD} \subset Topic_{LD}$$

in the following way:

$$LPerc_{LD} = \{\text{Int}(X \text{ is } \mathcal{A}_i) \mid \text{Int}(X \text{ is } \mathcal{A}_i)(w)(u_0) > 0$$

$$\text{and } \forall \text{Int}(X \text{ is } \mathcal{A}_j) \in Topic_{LD} :$$

$$((X \text{ is } \mathcal{A}_j) \preceq (X \text{ is } \mathcal{A}_i)) \Rightarrow ((X \text{ is } \mathcal{A}_j) = (X \text{ is } \mathcal{A}_i)) \quad (8)$$

Observe that $LPerc_{LD}$ depends on the input value u_0 (perception) as \preceq also depends on it. For every intension in $LPerc_{LD}$ we have value of its extension $b_i = (\text{Int}(X \text{ is } \mathcal{A}_i)(w))(u_0)$.

Then the PbLD might be summarized with the following deduction rule

$$r_{PbLD} : \frac{LPerc_{LD} = \{\text{Int}(X \text{ is } \mathcal{A}_i)\}, LD}{C_i(v) = b_i \rightarrow (\text{Int}(Y \text{ is } \mathcal{B}_i)(w'))(v)}, \quad (9)$$

which yields a fuzzy sets on w' . The rule produces projections of fuzzy relations defined for every rule \mathcal{R} whose antecedent is in $LPerc_{LD}$. In other words, only a subset of rules in LD fires

If the rule r_{PbLD} gives more than one fuzzy set then they are combined with the minimum operator:

$$C = \bigwedge_{i=1}^L C_i,$$

where L is the number of antecedents in $LPerc_{LD}$. The resulting fuzzy set is defuzzified by *last of maxima*, *first of maxima* and *mean of maxima* whether C is non-increasing, non-decreasing or increasing and decreasing respectively. The defuzzification function in PbLD is called *Defuzzification of evaluative expressions* (DEE) and its application on evaluative expressions can be seen in Figure 1.

2.5. Mining of IF-THEN rules from data

PbLD was designed to allow the experts who can not program to interact with a computer through formulating conditional clauses (fuzzy/linguistic IF-THEN rules). The experts are usually able to provide some linguistic description of a modeled problem. However, this is not always the case and thus data-mining methods for extracting such linguistic expressions has to be used. Below we describe an approach we used to find the fuzzy/linguistic IF-THEN rules for our multivariate times-series modeling. It is in the tradition of association mining [13] as originally studied under the name GUHA [14]. We use version of fuzzy association mining, more details can be found in [15] and its references.

Assume that we have in data variables X and Y . We would like to find linguistic description (3) of the dependency of Y on X . We choose a context $w = \langle v_L, v_M, v_R \rangle$ of a variable (e.g. X), where v_L (resp. v_R) is usually minimum (resp. maximum) of a variable and $v_M = \frac{v_L + v_R}{2}$, but it does not need to be. The chosen context w is used for every evaluative predication “ X is \mathcal{A} ” that we want to be possibly (we do not know yet if the data will support it) present in our model. Then for every value $x_i \in X$ and every evaluative predication “ X is \mathcal{A} ” we calculate:

$$\text{Int}(X \text{ is } \mathcal{A})(w)(x_i),$$

which are membership degrees to fuzzy sets representing particular linguistic predications. We replace variable X with the just calculated membership degrees and we get n new variables in our data, where n is the number of linguistic predications we are using in our model. We repeat the process for every variable in our data. We call the resulting data *linguistic description data*, LDD for short. We denote o an object (line) in our LDD data.

For brevity we will use the following notation: $X(o) = (\text{Int}(X \text{ is } \mathcal{A})(w))(o)$, $Y(o) = (\text{Int}(Y \text{ is } \mathcal{B})(w'))(o)$. Note that $X(o)$ is compound when the expression “ X is \mathcal{A} ” is compound. We define a *support* ($\text{supp}_{\otimes}(X \rightarrow Y)$) and *confidence* ($\text{conf}_{\otimes}(X \rightarrow Y)$) of a rule ($X \rightarrow Y$) in the following way:

$$\text{supp}_{\otimes}(X \rightarrow Y) = \frac{\sum_{o \in \text{LDD}} X(o) \otimes Y(o)}{n}, \quad (10)$$

$$\text{conf}_{\otimes}(X \rightarrow Y) = \frac{\sum_{o \in \text{LDD}} Y(o) \otimes X(o)}{\sum_{o \in \text{LDD}} X(o)}, \quad (11)$$

where \otimes is arbitrary t-norm. The aim is to search for such rules in LDD that have reasonably high confidence (e.g. above 0.8) and arbitrary support (e.g. from 0.05 and higher). The numbers will vary significantly based on the application. The found rules are then interpreted as fuzzy/linguistic IF-THEN rules and we obtain in this way some linguistic description LD.

We experimented with different t-norms. In various contexts the minimum t-norm was always giving the best model with respect to the final root mean square error. We do not know the reasons for that and do not claim that it will be like that always in every context. However, whether there is any optimal t-norm to be used will be further investigated.

We used software R [16] and particularly `lfl`: Linguistic fuzzy logic package [17] that is downloadable from CRAN for mining the linguistic descriptions. Also `PbLD` is implemented in `lfl` package. The parallel version of algorithm used for mining the fuzzy/linguistic rules is described in [18] and is based on algorithm `OPUS` [19].

3. Multivariate time series application

In this section we describe our methodology for modeling the multivariate time series. We assume we have n -variate time series X_1, \dots, X_n . The value of time series X_i in time t is denoted as $X_i(t)$. We do not assume any causality relationships. Very often the causality between two time series is in both directions, but just shifted in time.

We build our model of multivariate time series X_1, \dots, X_n in the following steps:

Step 0: We create a trivial data set by choosing one of the time series $X_i \in X_1, \dots, X_n$.

Step 1: We extend the data from Step 0 with derived variables. For every variable $X_i = X_1, \dots, X_n$ we define variables X'_i such as shifts to the past:

$$X_i^{S(h)}(t) := X_i(t - h),$$

where $h > 0^1$ is a shift in the time, differences of zero order:

$$X_i^{D(0,l,h)}(t) := X_i(t - h) - X_i(t - h - l),$$

where $l > 0$ is lag of a difference and $h > 0$ is shift to the past, differences of first order:

$$X_i^{D(1,l,h)}(t) := X_i^{D(0,l,h)}(t - h) - X_i^{D(0,l,h)}(t - h - l),$$

and moving averages:

$$X_i^{MA(n)}(t) := \frac{1}{n} \sum_{j=1}^n X_i(t - j),$$

Where n is the length of a moving average.

We may add also differences of higher orders, but more importantly time variable, which is not derived from any other variable:

$$X_{time}(t) := t.$$

The importance of adding the time variable will be explained later by equation (12).

Step 2: Data from Step 1 are used to build LDD data by a procedure described in Section 2.5.

Step 3: We mine fuzzy/linguistic IF-THEN rules from LDD data we get in Step 2 by a procedure described in Section 2.5 with restriction that we allow only the variable chosen at Step 0 to be present on the right hand side of the mined rules.

Step 4: We repeat the steps 0 through 3 to obtain set of fuzzy/linguistic IF-THEN rules for every X_i .

¹The h may be also negative (shifts into the future) for time series, which future we know - e. g., company production plan. Usually the h is based on periodicity and seasonality of time-series, but we may define various shifts in time $h = 1, \dots, n$ and let the data mining method in Step 3 to decide, which shift will be used in our model.

The sets of mined IF-THEN rules comprise our model of multivariate time series X_1, \dots, X_n . To obtain the values of particular time series, we give as an input to PbLD the mined IF-THEN rules together with data created in Step 1, which in this context plays a role of the local perception, see Section 2.4. To be able to test our model we have to hide part of the data created in Step 1. Moreover, on every line in data created in Step 1 we have to oblige to a rule that whenever we have values $X_i(t), X_i'(t'), X_j'(t'')$ then always $t', t'' < t$. In other words, we are hiding the present time. This is not needed in every multivariate time-series modeling (e.g. in marketing we know past, present and future of ad campaigns), but in our macroeconomic modeling it is crucial.

The inclusion of the time variable allows us to search also for the fuzzy/linguistic IF-THEN rules with the antecedents such as:

$$X_i(t-1) \text{ is } \textit{very small} \textbf{ AND } X_{time}(t) \text{ is } \textit{big}, \quad (12)$$

which means that previous value of variable X_i was small and current time is big (near the end of time series). Thus our model is also non-stationary.

3.1. Macroeconomic Example

We show an application of our methodology described in the previous section to the macroeconomic data that were obtained from Datastream database [20]. Our data consist of three time series: *The gap of economic output* (G), which is a logarithm of the difference between the trend of European GDP and its current level, *Interest Rate* (IR), and *Inflation* (I). The data are quarterly since 1985 till 2009, see Figure 2.

The derived variables we used in Step 2 of building our model were shifts to the past $X_i^{S(h)}$ with $h = 1, \dots, 12$, differences of zero and first order $X_i^{D(0,l,h)}, X_i^{D(1,l,h)}$ with $l = 4, 8$ and $h = 1, 2$, and moving averages $X_i^{MA(n)}$ with $n = 1, \dots, 12$.

Here is an example of a rule mined:

$$\mathbf{IF} \text{ IR}^{D(0,4,1)} \text{ is } \textit{Big} \textbf{ AND } \text{ IR}^{S(6)} \text{ is } \textit{Big} \quad (13) \\ \mathbf{THEN} \text{ G is } \textit{Big},$$

which says that if the Interest rate last quarter increased much and was high one and a half year ago then the Gap of economical output is Big.

In Step 3 we mined fuzzy/linguistic IF-THEN rules with the thresholds for support and confidence set as 0.05 and 0.8 respectively. The rules obtained with this setting were giving the optimal model with respect to the root mean square errors.

We compared our model with New-Keynesian DSGE model published in [21]. Our aim was to be at least comparable with DSGE model, but we even managed to outperform it in certain criteria. The root mean square errors of our Linguistic and DSGE model are shown in Table 3. Linguistic model is

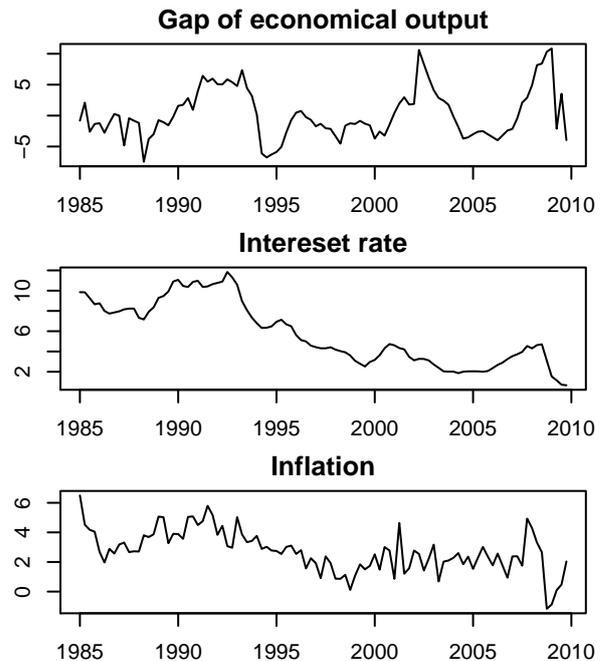


Figure 2: Macroeconomic time series

	Linguistic Model	DSGE Model
G	1.27	2.00
IR	0.64	0.50
I	0.86	0.93

Table 3: Root mean square errors of Linguistic and Classical economical models

modeling (predicting only one step ahead) two of three macroeconomic time series with smaller error than the DSGE model. An example of detailed graphical comparison on Gap of economical output is shown in Figure 3.

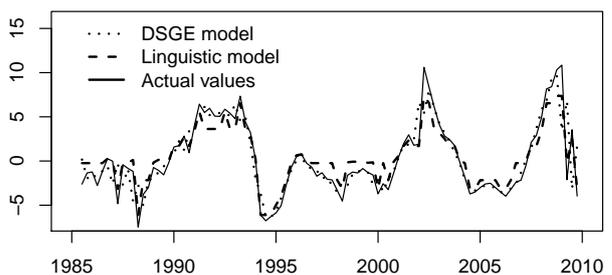


Figure 3: Linguistic and DSGE models of Gap of economical output

The main advantage of our model is that for every value modeled we may provide explanation as a set of IF-THEN rules as given in equation (13). The whole model may be too complex (e. g., above 1000 rules in linguistic description) and not be very understandable, but because in PbLD inference the rules are filtered out (see Equation (9)), we obtain only small subset of the fuzzy/linguistic IF-THEN rules (sometimes only one rule), while we model particular value in time series. These rules can be

straightforwardly translated into structured text of the conditional clauses in natural language and the user obtains easy to understand explanations.

4. Conclusion and future work

We have developed precise model of multivariate macro economical time series that relies only on simple IF-THEN rules. The advantage is that such rules are readable as simple conditional clauses in human language. Our future work will include discussions with the economists about these rules and we want to find out whether the rules contain obvious information or the economists may learn something from our model. The ability of our model to provide explanations in natural language will be in this context crucial.

One disadvantage of our approach is that there are many parameters to set up before we produce a model. We do not have presently a methodology to automatically deduce such parameters. Providing some methodology and automation in this area will be the focus of our future work.

The further usage of support and confidence measures in the inference along with the investigation of applicability of other fuzzy measures will be also the subject of the future research.

Acknowledgments.

This work was supported by the European Regional Development Fund in the IT4Innovations Center of Excellence project (CZ.1.05/1.1.00/02.0070).

References

- [1] Robert H Shumway and David S Stoffer. *Time series analysis and its applications: with R examples*. Springer, 2010.
- [2] George EP Box and Gwilym M Jenkins. *Time series analysis: forecasting and control, revised ed.* Holden-Day, 1976.
- [3] Xiaoyue Wang, Abdullah Mueen, Hui Ding, Goce Trajcevski, Peter Scheuermann, and Eamonn Keogh. Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery*, 26(2):275–309, 2013.
- [4] Martin Štěpnička, Paulo Cortez, Juan Peralta Donate, and Lenka Štěpničková. Forecasting seasonal time series with computational intelligence: On recent methods and the potential of their combinations. *Expert Syst. Appl.*, 40(6):1981–1992, 2013.
- [5] Martin Štěpnička, Antonín Dvořák, Viktor Pavliška, and Lenka Vavříčková. A linguistic approach to time series modeling with the help of f-transform. *Fuzzy Sets and Systems*, 180(1):164–184, 2011.
- [6] Irina Perfilieva. Fuzzy transforms: Theory and applications. *Fuzzy sets and systems*, 157(8):993–1023, 2006.
- [7] Vilém Novák. A comprehensive theory of trichotomous evaluative linguistic expressions. *Fuzzy Sets and Systems*, 159(22):2939–2969, 2008.
- [8] Lotfi A Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *Systems, Man and Cybernetics, IEEE Transactions on*, (1):28–44, 1973.
- [9] Lotfi A Zadeh. Precised natural language (pnl). *AI magazine*, 25(3):74, 2004.
- [10] Eva Hajicová, Barbara Hall Partee, and Petr Sgall. *Topic-focus articulation, tripartite structures, and semantic content*. Kluwer Academic, 1998.
- [11] Vilém Novák and Irina Perfilieva. On the semantics of perception-based fuzzy logic deduction. *International Journal of Intelligent Systems*, 19(11):1007–1031, 2004.
- [12] Vilém Novák. Perception-based logical deduction. In *Computational Intelligence, Theory and Applications*, pages 237–250. Springer, 2005.
- [13] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.
- [14] P Hájek and T Havránek. *Mechanizing Hypothesis Formation (Mathematical Foundations for a General Theory)*. Springer-Verlag, 1978.
- [15] Didier Dubois, Eyke Hüllermeier, and Henri Prade. A systematic approach to the assessment of fuzzy association rules. *Data Mining and Knowledge Discovery*, 13(2):167–192, 2006.
- [16] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.
- [17] Michal Burda. *lfl: Linguistic Fuzzy Logic (R package on CRAN)*, 2015. R package version 1.0.
- [18] Michal Burda, Viktor Pavliška, and Radek Valasek. Parallel mining of fuzzy association rules on dense data sets. In *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*, pages 2156–2162. IEEE, 2014.
- [19] Geoffrey I. Webb. Opus: An efficient admissible algorithm for unordered search. *Journal of Artificial intelligence Research*, 3:431–465, 1995.
- [20] THOMSON REUTERS. Datastream professional.
- [21] Martin Melecký and Aleš Melecký. *"Applied modelling for the need of economic policy" (in czech)*. Series on Advanced Economic Studies. VŠB-TU Ostrava, 2012.