Computation of a Sufficient Condition for System Input Redundancy

S. E. Papadakis and V. G. Kaburlasos

Technological Institute of Kavala
Department of Industrial Informatics
65404 Kavala, Greece
{spap,vgkabs}@teikav.edu.gr

Abstract. The calculation of an optimal subset of inputs from a set of candidate ones is known in the bibliography of system modeling as *the input (or feature) selection problem*. In this work we introduce a remarkable attribute of the FLR classifier: it's capacity to identify redundant system inputs, from a set of input/output data. The proposed approach is applicable beyond R^N on any lattice ordered data set $\mathbb{L}^N$, which may include disparate types of data. Also, the proposed approach can deal with populations of data instead of crisp data vectors. Finally, it is highlighted that proposed approach can be employed for designing models with simple structure and significant performance. The method is successfully applied here on two well known *real world* classification problems, identifying redundant inputs and inducing FLR classifiers with simple structure and favorable classification performance.

Key words: Fuzzy Interval Number (FIN), Fuzzy Lattice Reasoning (FLR), classification, lattice theory

1 Introduction

One of the most important issues in computational intelligence is the structure identification of the model to be used for modeling effectively a physical system. According to established modeling methodology, physical systems are treated as *black boxes* where the only knowledge we have about them emanates from a finite set of samples, which are organized as ordered pairs of input (excitations) output (response) data. The next step in modeling, given a sufficient set of input/output data, is the definition of proper model's structure and a learning rule. Since samples are always finite, an effective modeling must produce models with sound generalization capacities. That is, models which have as close as possible behavior to physical system, especially on data lying outside the initial set of samples.

The related bibliography on effective model's definition including fuzzy modeling [1–5], neural networks [6–8] , self organizing maps [9], mathematical models [10–12], etc is vast. In general, structure's identification problem can be hierarchically divided into three principal sub-problems. The first one, namely *input*

pp. 23–31, ISBN 978–80–244–2112–4, Palacký University, Olomouc, 2008.

selection problem, is the choice from a set of (intuitively selected) candidate inputs of those ones that are necessary and sufficient to describe the specified target. Second, is the formation of inner model structure (i.e. number of rules, or neurons, input space partition etc.) in the space of selected inputs. Third, is the process of parameter identification (training) by applying a convenient learning rule.

The importance of input selection problem has been widely recognized by many researchers [13, 14, 4] . For instance, in [3, 4] the authors claim that using a scale of importance varying from one to one hundred, the input selection problem is rated as one hundred, the inner model's structure is rated as ten, while the task of parameters identification is rated as one.

A detailed review of several input selection approaches is referred to in [13, 15]. Although the proposed method encounter all aforementioned three types of structure identification problems, the discussion in this work focuses mainly on input selection issues.

The basic advantage of our methodology, presented below, is that it can be applied on non-homogeneous input/output data including real numbers, populations of data represented by probability/possibility distributions,etc. Disparate types of data can be represented by Fuzzy Interval Numbers (FINs) [16–18]. The set of FINs is lattice ordered, where a metric can be defined by establishing a parametric positive valuation function [16]. Parametric positive valuation functions introduce several tunable non-linearities, which are adjusted by a stochastic non linear optimization method toward maximization of performance.

The layout of this preliminary work is as follows: Section 2 presents the mathematical background. Section 3 presents the proposed method. Section 4 presents experimental results. Finally, section 5 concludes by summarizing the contribution of this work.

2 Mathematical Background

In order to make the proposed approach clear, some notions are shown next. A *generalized interval* is denoted by $[x, y],\ x, y \in R$. Let $(\Delta, \leq) = (R, \leq^{\partial}) \times (R, \leq)$ be the complete product lattice of generalized intervals. Note that the inverse $\geq$ of any order relation $\leq$ is itself an order relation. The order $\geq$ is called the *dual* of $\leq$, symbolically $\leq^{\partial}$, or $\leq^{-1}$, or $\geq$.

The corresponding *meet* and *join* in lattice $(\Delta, \leq)$ are given by $[a, b] \wedge [c, d] = [a \vee c, b \wedge d]$ and $[a, b] \vee [c, d] = [a \wedge c, b \vee d]$. Note that $(\Delta, \leq) = (\Delta_-, \leq) \cup (\Delta_+, \leq)$ where $(\Delta_-, \leq)$, $(\Delta_+, \leq)$ is the set of negative $(a > b)$ and positive $(a \leq b)$ generalized intervals, respectively. Note that lattice $(\Delta_+, \leq)$ is isomorphic to lattice $(\tau(R), \leq)$ of intervals in set R, That is $(\tau(R), \leq) \cong (\Delta_+, \leq)$.

A strictly decreasing function $\theta_R : R \to R$ implies an isomorphism $(R, \leq) \cong (R, \geq)$. Furthermore, a strictly increasing function $\nu_R : R \to R$ is a positive valuation function in lattice $(R, \leq)$. Hence, function $\nu_\Delta : \Delta \to R$, given by $\nu_\Delta([a, b]) = \nu_R(\theta_R(a)) + \nu_R(b)$ is a positive valuation function in lattice $(\Delta, \leq)$ [19]. It follows a metric $d_\Delta : \Delta \times \Delta \to R_0^+$ given by

$$d_{\Delta}([a,b],[c,d]) = [\nu_R(\theta_R(a \wedge c)) - \nu_R(\theta_R(a \vee c))] + [\nu_R(b \vee d) - \nu_R(b \wedge d)] \quad (1)$$

A **G**eneralized **I**nterval **N**umber is a function $f(0,1] \rightarrow \Delta$ where Δ denotes a complete product lattice $(\Delta, \leq) = (R, \leq^{\partial}) \times (R, \leq)$ of generalized intervals.

2.1 Fuzzy Interval Numbers

Fuzzy Interval Numbers or FINs have been extensively described in [16], [17], [18]. Let $\mathbb{L}$ to be the Lattice of FINs. Then a N-tuple FIN $\mathbf{F} \in \mathbb{L}^N$. A FIN F can be represented as the union set of generalized intervals $F = \underset{h \in (0,1]}{\cup} \{[a_h, b_h]\}$. IF $a_h = b_h \forall h \in (0,1]$ the FIN is called *trivial FIN*. Given a strictly increasing function: $\nu(.)$ and a strictly decreasing one: $\theta(.)$ an inclusion measure is defined as a function $\sigma_{\mathbb{L}} : \mathbb{L} \times \mathbb{L} \rightarrow [0,1]$, given by:

$$\sigma_{\mathbb{L}}(\mathbf{F}, \mathbf{E}) = \int_0^1 \frac{\sum_{i=1}^{N} [\nu_{R,i}(\theta_{R,i}(c_{i,h})) + \nu_{R,i}(d_{i,h})]}{\sum_{i=1}^{N} [\nu_{R,i}(\theta_{R,i}(a_{i,h} \wedge c_{i,h})) + \nu_{R,i}(b_{i,h} \vee d_{i,h})]} dh \quad (2)$$

Note that $\mathbf{F}, \mathbf{E}$ are N-tuple FINs. Functions $\nu : R \rightarrow R_0^+$ and $\theta : R \rightarrow R$ can be given by:

$$\begin{aligned} \nu_{R,i}(x) &= \tfrac{A_i}{1+e^{-\lambda_i(x-m_i)}} \\ \theta_{R,i}(x) &= 1 - 2m_i \end{aligned} \quad (3)$$

where $i = 1, 2, ..., N$. N denotes the number of inputs. $A_i, \lambda_i, m_i \in R$ tunable parameters. The size of a FIN is defined as a function $Z_F : \mathbb{F} \times \mathbb{F} \rightarrow R_0^+$, given by:

$$Z_F(\mathbf{F}) = \int_0^1 d_{\Delta}(a_h, b_h) dh \quad (4)$$

where $d_{\Delta}(a_h, b_h)$ is computed by Eq. (1)

3 The Proposed Method

The granular FLR is a set of labeled pairs $(\mathbf{E}_i, c_i)$ or *granules* , each represented by a N-tuple FIN $\mathbf{E}_i$ and a label c_i. Linguistically, a granule defines a rule of the form: *IF datum* $\boldsymbol{F}_j$ *is included in granule* $\boldsymbol{E}_i$ *then the class of datum* $\boldsymbol{F}_j$ *is* c_i. Hence a set $RB = \{E_i, c_i\}$ of granules defines a rule base for the FLR classifier.

The FLR algorithm is divided in two parts:

Algorithm 1 FLR for training

1: Initialize $RB = \{\mathbf{E}_\ell, c_\ell\} \mid \ell = 1, 2, ..., L$ of granules $\mathbf{E}_\ell$. $c_\ell \in C$ is the label of granule $\mathbf{E}_\ell$.
2: Do *set* all pairs in RB. Present the next input pair $(\mathbf{F}_i, c_i)$, i=1,...,n. Compute the degree of inclusion $\sigma_F(\mathbf{F}_i \leq \mathbf{E}_\ell)$ of input granule $\mathbf{F}_i$ to all granules $\mathbf{E}_\ell$, $\ell = 1, ..., L$.
3: IF no pairs are *set* in RB then store input pair $(\mathbf{F}_i, c_i)$ in RB, $L = L + 1$, goto 2.
4: The winner among *set* pairs in RB is $\mathbf{E}_j, c_j$ such that $j = \underset{\ell \in \{1,...L\}}{\arg\max} \{\sigma_F(\mathbf{F}_i \leq \mathbf{E}_\ell)\}$.
5: *The Assimilation Condition*: Both 1.) The size $Z(\mathbf{F}_i \vee \mathbf{E}_j)$ of granule $\mathbf{F}_i \vee \mathbf{E}_j$ is less than a user defined threshold size Z_{crit} and 2.) $c_i = c_j$.
6: if the *Assimilation Condition* is not satisfied then *reset* the winner and goto 3. Else, replace the winner with granule $\mathbf{F}_i \wedge \mathbf{E}_j$; and goto 2.

Algorithm 2 Stochastic optimization of tunable FLR parameters

1: Select a population of individuals and encode $Z_{crit}, A_i, \lambda_i, m_i$ where $i = 1, 2, ..., N$ into chromosome.
2: For each individual apply algorithm 1 and calculate its Fitness given by Eq. (5), (see Section 4).
3: Apply genetic operators to produce next generation.
4: if stopping criterion is not satisfied then goto 2.
5: Store the trained FLR, consisting of a RB of labeled granules and fine tuned parameters $Z_{crit}, A_i, \lambda_i, m_i$.

The results of algorithm 1 depend on both Z_{crit} and parameters $A_i, \lambda_i, m_i \,| i = 1, ..., N$ according to Eq. (2), (3). The tunable parameters Z_{crit} and parameters A_i, λ_i, m_i are optimized such that the rate of success classification is maximized.

In the case where $\nu_{R,i}(x) = const_i \; \forall \; x, i \mid const_i \in R$, it follows by equations 1, 2 that any calculated distances and similarity measures take constant values for every data $\mathbf{F}_j, \mathbf{F}_{k \neq j}$. Hence, there is no any discretization information and all data are equally distant (or equally similar) to each other. The classification process has to be entirely based only on the attributes (if any) with respective non flat sigmoid positive valuation functions. In other words attributes which have constant sigmoid positive valuation functions in the range of their interest, provides no discretization information and may be omitted. The aforementioned reasoning is experimentaly verified next.

4 Experimental Results

It has to be stressed that here we use input/output data, which are real numbers. However, the method has been developed using FINs data representation, which are members of a lattice. Hence, without loss of generality, a FIN represents here a real number.

4.1 Fisher's IRIS Classification Benchmark

Fisher Iris benchmark data set, downloaded from the UCI machine learning repository [20], is used to demonstrate the proposed feature selection technique. It includes measurements regarding four crinum flower attributes including x_1:*sepal-length*, x_2:*sepal-width*, x_3:*petal-length*, x_4:*petal-width*. The crinums are classified in three classes, namely *versicolor* (i.e. class 1), *setosa* (i.e. class 2), and *virginica* (i.e. class 3). In all, there are fifty 4-dimensional vectors per class available. After a random data permutation we employed the first 50 data vectors (33.33%) for training the next 50 (33.33%) for validation and the last 50 (33.33%) for testing. Each input datum is considered as a 4-tuple trivial FIN. Each sigmoid $\nu_i(x) \mid i = 1, 2, 3, 4$, is defined by three tunable parameters A_i, λ_i, m_i.

The genetic algorithm, presented in [21], is used for the optimization of FLR. The optimization of FLR lies in the calculation of sigmoid's parameter such that the rate of successful classification on both training and validation data set is as high as possible. Sigmoid parameters are binary encoded into the chromosome of every individual, using a word of 16 bits per parameter. Each individual represents a FLR model which is created using the encoded parameters' value, and applying the algorithm 2 on the training data set. Hence, the percent classification rate on both training and validation data set, namely (R_{trn}) and (R_{val}) respectively, is calculated. The fitness function is calculated by Eq. (5)

$$Q(\mathbf{p}_s) = w \cdot R_{trn} + (1 - w) \cdot R_{val} + \frac{0.1}{L} \tag{5}$$

The parameter $\mathbf{p}_s = [Z_{crit}, A_1, \lambda_1, m_1, ..., A_N, \lambda_N, m_N]$ denotes the vector of tunable parameter values, which are encoded into the chromoshome of individual $s, s = 1, 2, ...S \mid S$ denotes the population size. For N attributes the vector $\mathbf{p}_s$ has $3N + 1$ elements. Parameter L is a positive integer, which depends on the value of parameter Z_{crit} and denotes the number of granules in RB that constitute the FLR model. Term $0.1/L$ in Eq. (5) is used to lead the evolution into FLR models with RB having small number of granules. Finally, $w = 0.5$ is a relaxation factor, used to direct the evolution out of over trained solutions. The GA population includes 25 individuals and the evolution terminates when the quality function of the elite individual remains intact for 50 successive generations.

The sigmoid functions of the trained FLR are illustrated in Fig. 1. Stated by experimental results, we conclude that attribute *sepal width* is negligible, since all *sepal width* input values are mapped to a constant value approximately equal to 2. As a result attribute *sepal width* does not provides any class discretization information and should be removed. Our claim was experimentally verified by recalculating the classification rate without using *sepal width* values. We remarked that ignorance of *sepal width* does not affect training, validation and checking classification rate.

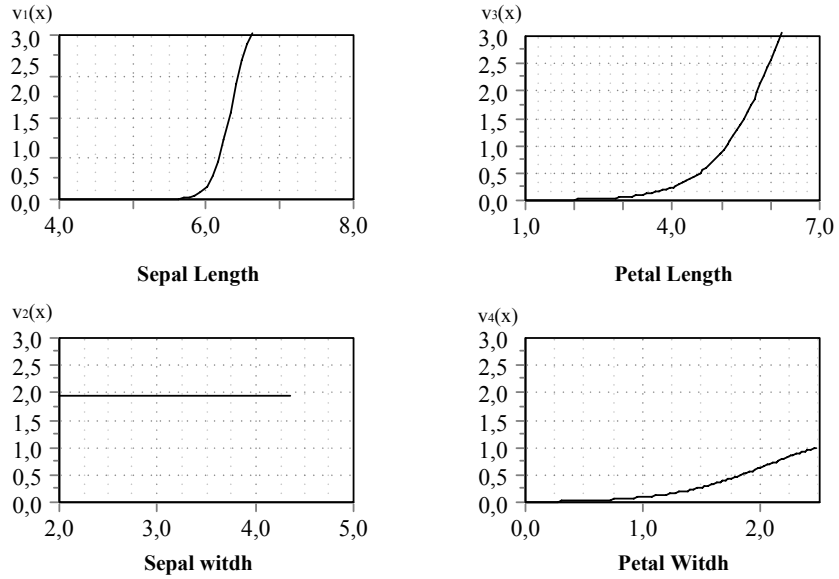

Fig. 1. The sigmoid positive valuation functions of the four attributes for the Fisher's IRIS classification benchmark.Since the sigmoid function for *Sepal width* is flat we conclude that the corresponding input is redundant.

4.2 The Wine Classification Benchmark

A more complex problem, the well known wine classification benchmark, is used in this example for further verification of proposed approach. In this problem is faced the classification of wines, in three categories according to 13 measured chemical attributes. Each input datum is created as a 13-tuple trivial FIN. Each datum's output takes an integer value 1,2, or 3 which represents the class of specific wine sample. The total of 178 input - output data were separated in three subsets, namely, training (60 data), validation (60 data) and testing (58 data) set. Algorithm FLR was applied using all 13 attributes. After training, the sigmoid functions of attributes are illustrated in Fig. 2. It is clearly observed that six attributes: Alcohol, Alkalinity of Ash, Magnesium, Total phenols, Nonflavonoid phenols and Proanthocyanins illustrate flat sigmoid in the range of interest. Thus, they are redundant. Ignoring redundant attributes a simplified FLR model classifier was built, which provides the same classification rate with FLR created by all thirteen attributes. This result confirms experimentally the statement that specific attributes are unnecessary.

5 Discussion and Conclusion

A novel and effective method for the determination of redundant attributes in classification problems was introduced. Our proposed method for input selection

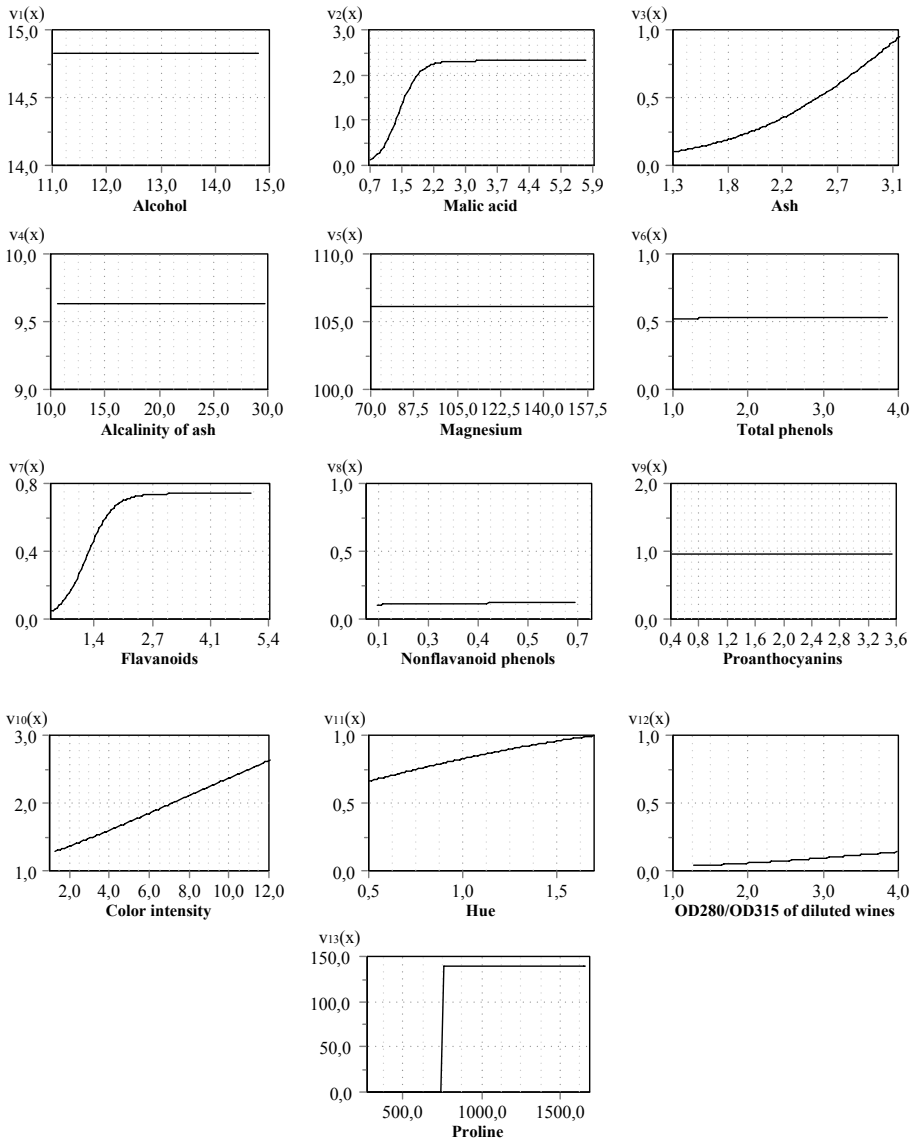

Fig. 2. The wine Classification Benchmark. Attributes with flat sigmoid positive valuation functions are redundunt ones since all their input values are mapped to a constant number.

using the FLR classifier computes only *sufficient* conditions. Hence when the sigmoid (positive valuation function) is constant in the range of interest of an input variable then we conclude that the latter input variable is redundant and can be omitted. In other words, for not-constant positive valuation functions we cannot tell whether the corresponding input variable is redundant or not. More work needs to be done in this direction in the future. Considering that we need only three tunable parameters per classifier's input, we conclude that the proposed methodology produces FLR classifiers with simple structure. Moreover, the proposed technique is applied on data sets which are Lattices. In a future work our proposed method will be applied on disparate types of data as symbols and populations of measurements represented by probability distributions.

References

1. Papadakis, S., Theocharis, J.: A GA-based fuzzy modeling approach for generating TSK models. Fuzzy Sets and Systems **131** (2002) 121–152
2. Simpson, P.: Fuzzy min-max neural networks -part i: Classification. IEEE Transactions on Neural Networks **3** (1992) 776–786
3. Sugeno, M., Tanaka, K.: Successive identification of a fuzzy model and its applications to prediction of a complex system. Fuzzy Sets and Systems **42** (1989) 315–334
4. Sugeno, M., Yasukawa, T.: A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1** (1993) 7–31
5. Wang, X., Baets, B., Kerre, E.: A comparative study of similarity measures. Fuzzy Sets and Systems **73** (1995) 259–268
6. Chakraborty, D., Pal, N.: A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. IEEE Transactions on Neural Networks **15** (2004) 110–123
7. Patra, J., Pal, R., Chatterji, B., Panda, G.: Identification of nonlinear dynamic systems using functional link artificial neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B **29** (1999) 254–262
8. Rost, B., Sander, C.: Prediction of protein secondary structure at better than 70% accuracy. Journal of Molecular Biology **232** (1993) 584–599
9. Kohonen, T., Somervuo, P.: How to make large self-organizing maps for nonvectorial data. Neural Networks **15** (2002) 945–952
10. Lichstein, J., Simons, T., Shriner, S., Franzreb, K.: Spatial autocorrelation and autoregressive models in ecology. Ecological Monographs **72** (2002) 445–463
11. Rabiner, Lawrence, R., Juang, Biing-Hwang: Introduction to hidden markov models. IEEE ASSP magazine **3** (1986) 4–16
12. Zhang, B.: An EM algorithm for a semiparametric finite mixture model. Journal of Statistical Computation and Simulation **72** (2002) 791–802
13. Arauzo-Azofra, A., Benitez, J., Castro, J.: Consistency measures for feature selection. Journal of Intelligent Information Systems **30** (2008) 273–292
14. Ivakhnenko, A., Ivakhnenko, N.: Long-term prediction of random processes by gmdh algorithms using the unbiasedness criterion and balance-of-variables criterion. Soviet Automation and Control **7** (1974) 40–45
15. Papadakis, S., Tzionas, P., Kaburlasos, V., Theocharis, J.: A genetic based approach to the Type I structure identification problem. Informatica **16** (2005) 365–382

16. Kaburlasos, V.: FINs lattice theoretic tools for improving prediction of sugar production from population of measurements. IEEE Transactions on System Man and Cybernetics, Part-B **34** (2004) 1017–1030
17. Kaburlasos, V., Papadakis, S.: Granular self organizing map (grSOM) for structure identification. Neural Networks **19** (2006) 623–643
18. Papadakis, S., Marinagi, C., Kaburlasos, V., Theodorides, M.: Estimation of industrial production using the granular self-organizing map (grSOM). In: Proceedings of the 12th Mediterranean Conference on Control and Automation (MED'04), Kusadasi, Turkey (2004) session TuM2-D,proceedings in CD-ROM.
19. Kaburlasos, V., Athanasiadis, I., Mitkas, P.: Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. International Journal of Approximate Reasoning **45** (2007) 152–188
20. Newman, D., Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases. (1998)
21. Kazarlis, S., Papadakis, S., Theocharis, J.B., Petridis, V.: Micro-genetic algorithms as generalized hill-climbing operators for GA optimazation. IEEE Transactions on Evolutionary Computation **5** (2001) 204–217