STUDIA INFORMATICA 2008
Volume 29 Number 4A (80)

Krzysztof SIMIŃSKI
Politechnika Śląska, Instytut Informatyki

NEURO-FUZZY SYSTEM WITH HIERARCHICAL PARTITION OF INPUT DOMAIN

Summary. The paper presents the method of hierarchical domain partition in fuzzy inference system with parameterized consequences. The novelty of the solution is the partition based on fuzzy clustering. The experimental results on the synthetic and real life data set are also presented.

Keywords: fuzzy inference system, hierarchical domain partition

SYSTEM NEURONOWO-ROZMYTY Z HIERARCHICZNYM PODZIAŁEM PRZESTRZENI WEJŚCIOWEJ

Streszczenie. Artykuł przedstawia metodę hierarchicznego podziału dziedziny w systemie neuronowo-rozmytym. Nowością jest zastosowanie grupowania rozmytego do uzyskania podziału. Zaprezentowane także zostały wyniki eksperymenttów zarówno na syntetycznych, jak i na rzeczywistych zbiorach danych.

Słowa kluczowe: system neuronowo-rozmyty, hierarchiczny podział dziedziny

1. Introduction

In real life there is an ardent need to proceed the data when the explaining theories are lacking. Often only rules based on the experience of an expert are available. These are commonly based on linguistic variables and fuzzy approach can handle them better than crisp one. The second advantage of the neuro-fuzzy approach is the ability to induce knowledge from the presented data and the ability of knowledge generalization. Many neuro-fuzzy systems have been proposed, as ANFIS [7], HNFQ [22], NEFCLASS [12], NEFPROX [13], ANNBFIS [6, 9, 10], ANBLIR [8], Fuzzy RuleNet [23], FLEXNFIS [20].

In neuro-fuzzy systems the rule base is crucial. The premises of the rules split the domain into regions. Some methods have been elaborated to split the domain, among them the following can be mentioned: clustering, grid split or hierarchical split [8, 22].

The grid split created regions by splitting each attribute values into several intervals. The disadvantage of this approach is the curse of dimensionality – the number of regions grows exponentially with the number of input space dimension. Further some regions may be empty – containing no objects.

The domain partition achieved by clustering the input space avoids the curse of dimensionality, but there may occur the areas of very low membership to all regions.

The hierarchical partition of the domain has the advantages of the clustering and grid split. There is no curse of dimensionality in this approach and the areas with low membership to all regions are reduced.

Some attempts have been made to apply the hierarchical domain in neuro-fuzzy systems (LOLIMOT [14], binary space partitioning [21, 22, 1] – regions always split into two equal subregions). This paper depicts the attempt at applying the hierarchical approach to splitting the domain into regions in a fuzzy inference system with parameterized consequences.

2. Fuzzy Inference System with Parameterized Consequences

The system presented in [6, 9, 10] is the MISO system. The rule can be described as

$$R^{(i)} : \underline{X} \quad is \quad \underline{A}^{(i)} \Rightarrow Y \quad is \quad B^{(i)}(\underline{\theta}), \tag{1}$$

where $\underline{X} = [X_1, X_2, \ldots, X_N]^T$ and Y are linguistic variables, A and B are fuzzy linguistic terms (values) and $\underline{\theta}$ is the parameter vector of the consequent linguistic term.

The linguistic variable A is described with the Gaussian membership function:

$$\mu_A(x) = \exp\left(-\frac{(x-c)^2}{2\sigma^2} \right), \tag{2}$$

where c is the core location and σ is Gaussian bell deviation. Each region in the domain is represented by a fuzzy cluster with Gaussian membership function.

The term B is represented as an isosceles triangle with the base width w, the altitude of the triangle in question equal to the firing strength of the rule:

$$F^{(i)}(\underline{x}) = \mu_{A_1^{(i)}}(x_1) \bullet_T \ldots \bullet_T \mu_{A_N^{(i)}}(x_N), \tag{3}$$

where $\bullet_T$ denotes the T-norm. The localization of the core of the triangle membership function is determined by linear combination of input attribute values.

The fuzzy output of the system can be written as [5]

$$\mu_{B'}(y,\underline{x}) = \bigoplus_{i=1}^{I} \mu_{B^{(i)}}(y) = \bigoplus_{i=1}^{I} \Psi\left(\mu_{\underline{A}^{(i)}}(\underline{x}), \mu_{B^{(i)}}(y,\underline{x}) \right), \tag{4}$$

where $\oplus$ denotes the aggregation and Ψ the fuzzy implication. The crisp output of the system is calculated using the MICOG method [6]:

$$y = \frac{\sum_{i=1}^{I} g\left(F^{(i)}\left(\underline{x}, w^{(i)}\right)\right) y^{(i)}(\underline{x})}{\sum_{i=1}^{I} g\left(F^{(i)}\left(\underline{x}, w^{(i)}\right)\right)}, \tag{5}$$

where $y^{(i)}(\underline{x})$ stands for the location of center of gravity of the consequent fuzzy set, $F^{(i)}$ – the firing strength of the i-th rule, $w^{(i)}$ – the width of the base of the isosceles triangle consequence function. The function g depends on the fuzzy implication, in this paper the Reichenbach one is used, so [5]

$$g = \frac{w^{(i)}}{2} F^{(i)}\left(\underline{x}\right). \tag{6}$$

3. Hierarchical Domain Split

In the proposed approach the domain is hierarchically split. The algorithm is executed in three stages: (1) the selection of the worst region, (2) its split into two subregions and (3) the tuning of all regions.

In the first iteration there is only one region, so the selection of the worst region and its split is omitted and directly the tuning procedure is started. It is done so because the tuning is not only responsible for the better evaluation of the regions parameters, but it is necessary for the calculation of the parameters of the consequences. Having tuned the regions both the premises and the consequences of the rules are known and it is possible to elaborate the answer for input data.

The tuning of the region aims at better evaluation of region parameters and the elaboration of consequents. The parameters (cf. eq. 2) of the regions (the rules promises) and the widths of the bases of the isosceles triangle sets in consequences are tuned by means of the gradient method [6]. The linear coefficients of the localization of the triangle sets in consequence are calculated using the least square method [6].

The tuning enables the calculation of the system error for each data tuple. The region with the highest contribution to the root mean square error value for all N tuples is selected. The contribution of the rule's error to the global one is calculated with the following formula:

$$e_i = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left[g\left(F^{(i)}\left(\underline{x}_n, w^{(i)}\right)\right)\left(y^{(i)}(\underline{x}_n) - y_n\right)\right]^2} . \qquad (7)$$

The next stage is the splitting of the worst region. Having selected the worst one the splitting procedure is started. In the papers on the hierarchical split of domain in data mining there are described many methods using various criteria (e.g. Gini index [4], entropy [19, 24], impurity, information gain [11, 16], twoing rule [4], histogram [2], variance of the decision attribute [17, 18]). These methods are used for obtaining the crisp split of the domain. The second disadvantage of the above approaches is the testing of all possible splits of the domain what easily leads to the exponential increase of number of needed tests with adding more dimensions.

The fuzzy splitting of the domain is not very widely discussed in papers. Often the region is split by half [15, 1]. In our approach fuzzy c-means (FCM) algorithm is adapted for the region split. The worst region is clustered into two subregions. The problem appears which data tuples (objects) belong to the region to be split. This problem is solved in the following way. First the cardinality of the region is calculated (as a sum of membership values of all data tuples). The floor of the cardinality denotes the number of data tuples with the highest membership that will be treated as elements of the region in its splitting. If the floor of the region cardinality is less then one then the algorithm iterations loop breaks and the procedure of hierarchical domain partition ends. In clustering all attributes are taken into account, also the predicted attribute and the error of the system for each rule are added as additional attributes. The clustered domain can be written symbolically as $X \times E \times Y$, where X denotes the attribute space, E – the error space and Y – the output attribute space. The split by clustering gives the fuzzy partition of the region and avoid the curse of dimensionality. As the rules in our system are orthogonal the splitting attribute has to be selected. The attribute (dimension) with maximum value of index

$$s_i = \left|c_{1i} - c_{2i}\right| - s_{1i} - s_{2i} \qquad (8)$$

is selected as a splitting attribute (c_{1i} is the value of the *i*-th attribute in the center of the first cluster, s_{1i} is the standard deviation for this attribute in the first cluster). Only the parameters of the splitting attribute in the child regions are modified, the others are the same as in the parent region.

4. Experiments

The experiments were performed using the synthetic and real life data sets.

4.1. Carbon dioxide concentration in a pump deep shaft

The dataset contains the real life measurements of some air parameters in a pump deep shaft in one of Polish coal mines. The parameters (measured in one minute intervals) are: CO2 – the concentration of the carbon dioxide, Ps – atmospheric pressure, RHOs – relative humidity of the air in the shaft, RHPs – relative humidity of the air near the pump, TOs – air temperature. The dynamic attributes (the 10-minute sums of measurements – DCO2, DPs, DRHOs, DRHPs, DTOs) are added to the tuples. The task is to predict the concentration of the carbon dioxide in 10 minutes. The data are divided into train set (1828 tuples) and test set (915 tuples).

4.2. Methane concentration in the coal mine shaft

The dataset contains the real life measurements of some air parameters in a coal mine. The parameters (measured in 10 second intervals) are: AN31 – the flow of air in the shaft, AN32 the flow of air in the adjacent shaft, MM32 – methanometer, production of coal, the day of week, DAN31 – the sum of AN31 measurement in the last 10 minutes, DAN32 – 10-minute sum of AN32, DMM32 – 10-minute sum of MM32. The task is the prediction of the MM32 in 10 minutes. The data are divided into train set (499 tuples) and test set (523 tuples).

4.3. Boston housing

Benchmark Boston housing dataset contains the data about house prices in Boston. The attributes are: crime rate, proportion of residential land zoned for lots over 25,000 square feet, proportion of non-retail business acres, Charles River variable (1 if tract bounds river; 0 otherwise), nitric oxides concentration, average number of rooms per dwelling, proportion of owner-occupied units built prior to 1940, weighted distances to five Boston employment centers, index of accessibility to highways, full-value property-tax rate per \$10000, pupil-teacher ratio, $B = 1000(b - 0.63)^2$ (where b is the proportion of blacks by town), lower status of the population. The predicted value is median value of owner-occupied homes in \$1000's.

The dataset from *http://www.cs.toronto.edu/~delve/data/boston* was divided into train set (458 tuples) and test set (48 tuples).

4.4. Concrete Compressive Strength

The compressive strength of concrete is highly nonlinear function of ingredients and age. The dataset contains 1030 tuples with following attributes: cement (component 1) [kg/m3], blast furnace slag (component 2) [kg/m^3], fly ash (component 3) [kg/m^3], water (component 4) [kg/m^3], superplasticizer (component 5) [kg/m^3], coarse aggregate (component 6) [kg/m^3], fine aggregate (component 7) [kg/m^3], age (in days, 1–365). The predicted value is the concrete compressive strength in MPas.

The dataset from *http://archive.ics.uci.edu/ml/datasets/Concrete+Compressive+Strength* was split into train (tuples: 1 – 515) and test (516 – 1030) sets.

4.5. Chaotic time series prediction

The data set contains the data on concentration x of leukocytes in blood described with the Mackey-Glass equation

$$\frac{dx(t)}{dt} = \frac{ax(t-\tau)}{1+(x(t-\tau))^{10}} - bx(t), \tag{9}$$

where a, b and τ are constants. The data set containing 1000 tuples was split into train (1 – 500) and test (501 – 1000) sets.

4.6. Human Operation at a Chemical Plant

Benchmark data set without more detailed information downloaded from site: *http://neural.cs.nthu.edu.tw/jang/benchmark/*. The first half (tuples: 1–35) acted as a train set, the rest (tuples: 36 – 70) – test set.

4.7. Sombrero

It is a synthetic data set whose tuples represent the two-parameter function described with formula:

$$z = \frac{\sin\sqrt{x^2+y^2}}{\sqrt{x^2+y^2}}. \tag{10}$$

Train set (8963 tuples), test set (1037 tuples).

4.8. Gas furnace

This is a real life dataset depicting the concentration of methane and carbon dioxide in the gas furnace used by [3] and many researchers. It contains 290 tuples organized in the following manner

$$[\underline{x}, y] = [y(n-1), \ldots, y(n-4), x(n-1), \ldots, x(n-6), y(n)]. \tag{11}$$

5. Results

The presented approach was compared with other neuro-fuzzy systems in knowledge generalization.

Table 1
Comparison of RMSE errors obtained by ANNBFIS and proposed system

Dataset	Rules	ANNBFIS	Proposed system
CO_2 concentration	10	0.2818	0.1962
CH_4 concentration	10	0.7485	0.6553
Boston housing	10	1.4197	1.1516
concrete compressive strength	2	10.501	9.348
chaotic time series	10	0.002744	0.002459
human operations	10	378.29	331.21
sombrero	10	0.005402	0.003343

Table 2
Comparison of RMSE errors obtained by various systems and the proposed system for gas furnace data set

Author	Rules	RMSE
Tong	19	0.6848
Xu-Lu	25	0.5727
Pedrycz	81	0.5656
Box-Jenkins	1	0.4494
Sugeno-Yasukawa	6	0.4348
Chen	3	0.2678
Lin-Cunningham	4	0.2664
Sugeno-Tanaka	2	0.2607
Wang-Langari	2	0.2569
Zikidis-Vasilakos	6	0.2530
Kim-Park-Ji	2	0.2345
Kim-Park	2	0.2190
ANBLIR (Goedel)	2	0.1892
ANNBFIS	3	0.1791
proposed system	6	0.1681
ANNBFIS	6	0.1537

The abbreviation RMSE stands for “root mean square error”. The ANNBFIS system [6, 9, 10] splits the domain using the fuzzy clustering technique.

The results for “gas furnace” data set are compared with the achievements of the researchers who used this benchmark data set. The set is obviously organized in the same way to compare the results. The table 2 is taken from [5] with minor modification.

6. Conclusion

The hierarchical input domain partition joins the advantages of grid split and clustering. It lacks the faults of the grid split, that is the splitting of the input domain without regarding the data structure, what may lead to empty regions. Further it eradicates the exponential increase of region number with the increase of task dimensionality (the curse of dimensionality). This approach reduces also the existence of areas of very low membership to all regions, what may occur in clustering.

The neuro-fuzzy system for which the hierarchical partition of the input domain was proposed is the MISO system based on the neural network with parameterized consequences.

The hierarchical partition algorithm selects the worst region, ie. the region with the highest contribution to the global system error, splits the region in question into two subregions which replace the parent region. After splitting the region, all regions are tuned. This enables the better evaluation of region parameters and the elaboration of consequences.

The results of the experiments on the real life and synthetical datasets reveal that the hierarchical partition of the domain in the neuro-fuzzy systems can get better generalization ability.

REFERENCES

1. Almeida M. R. A.: Sistema hibrido neuro-fuzzy-genetico para mineracao automatica de dados. Master’s thesis, Pontifica Universidade Católica do Rio de Janeiro, 2004.
2. Basak J., Krishnapuram R.: Interpretable hierarchical clustering by constructing an unsupervised decision tree. IEEE Transactions on Knowledge and Data Engineering, 17(1), 2005, p. 121÷132.
3. Box G. E. P., Jenkins G.: Time Series Analysis, Forecasting and Control. Holden-Day, Incorporated, 1976.
4. Breiman L., Friedman J. H., Olshen R. A., Stone C. J.: Classification and Regression Trees. Wadsworth, Belmont 1984.

5. Czekalski P.: Evolution-fuzzy rule based system with parameterized consequences. International Journal of Applied Mathematics and Computer Science, 16 (3), 2006, p. 373÷385.
6. Czogała E., Łęski J.: Fuzzy and Neuro-Fuzzy Intelligent Systems. Series in Fuzziness and Soft Computing. Physica-Verlag, A Springer-Verlag Company, 2000.
7. Jang J.-S. R.: ANFIS: Adaptive-Network-Based Fuzzy Inference System. IEEE Transactions on Systems, Man, and Cybernetics, 23, 1993, p. 665÷684.
8. Łęski J.: Systemy neuronowo-rozmyte. WNT, Warszawa, 2008.
9. Łęski J., Czogała E.: A new artificial neural network based fuzzy inference system with moving consequents in if-then rules and selected applications. BUSEFAL, 7, 19971, p. 72÷81.
10. Łęski J., Czogała E.: A new artificial neural network based fuzzy inference system with moving consequents in if-then rules and selected applications. Fuzzy Sets and Systems, 108(3), 1999, p. 289÷297.
11. Murthy S. K., Kasif S., Salzberg S.: A system for induction of oblique decision trees. Journal of Artificial Intelligence Research, 2, 1994, p. 1÷32.
12. Nauck D., Kruse R.: NEFCLASS – A Neuro-Fuzzy Approach for the Classification of Data. Proceedings of the 1995 ACM Symposium on Applied Computing, 1995, p. 461÷465.
13. Nauck D., Kruse R.: Neuro-fuzzy systems for function approximation. Fuzzy Sets and Systems, 101, 1999, p. 261÷271.
14. Nelles O., Isermann R.: Basis function networks for interpolation of local linear models. Proceedings of the 35th IEEE Conference on Decision and Control, 1, 1996, p. 470÷475.
15. Nelles O., Fink A., Babuska R., Setnes M.: Comparison of two construc-tion algorithms for Takagi-Sugeno fuzzy models. International Journal of Applied Mathematics and Computer Science, 10 (4), 2000, p. 835÷855.
16. Quinlan J. R.: Induction of decision trees. Machine Learning, 1, 1986, p. 81÷106.
17. Quinlan J. R.: Learning with continuous classes. In Adams & Sterling, editor, AI'92, Singapore 1992, p. 343÷348.
18. Quinlan J. R.: Combining instance-based and model-based learning. ML'93, San Mateo 1993.
19. Rastogi R., Shim K.: PUBLIC: A decision tree classifier that integrates building and pruning. Data Mining and Knowledge Discovery, 4 (4), 2000, p. 315÷344.
20. Rutkowski, L., Cpalka K.: Flexible neuro-fuzzy systems. IEEE Transactions on Neural Networks, 14 (3) , 2003, p. 554÷574.

21. de Souza F. J., Vellasco M. B. R., Pacheco M. A. C.: Load forecasting with the hierarchical neuro-fuzzy binary space partitioning model. International Journal on Computer Systems and Signals, 3(2), 2002, p.118÷132.
22. de Souza F. J., Vellasco M. B. R., Pacheco M. A. C.: Hierarchical neuro-fuzzy quadtree models. Fuzzy Sets and Systems, 130(2), 2002, p.189÷205.
23. Tschichold-Gürman N.: Generation and improvement of fuzzy classifiers with incremental learning using fuzzy RuleNet. SAC '95: Proceedings of the 1995 ACM symposium on Applied Computing, 1995, p. 466÷470.
24. Wang Y., Witten I. H.: Inducing model trees for continuous classes. In Proc. of Poster Papers, 9th European Conference on Machine Learning, Prague, Czech Republic, April 1997.

Recenzent: Prof. dr hab. inż. Jacek Łęski

Wpłynęło do Redakcji 30 lipca 2008 r.

Omówienie

Artykuł prezentuje próbę zastosowania hierarchicznego podziału dziedziny w systemie neuronowo-rozmytym. Idea hierarchicznego podziału łączy w sobie cechy podziału siatkowego i grupowania. Nie posiada jednak wady podziału siatkowego polegającej na tworzeniu podziału dziedziny niezależnego od struktury danych. Unika się tutaj także niekorzystnego wykładniczego wzrostu liczby regionów wraz z liczbą wymiarów danych, które to zjawisko obserwuje się przy podziale siatkowym.

System neuronowo-rozmyty, dla którego został opracowany hierarchiczny podział dziedziny, jest systemem typu MISO opartym na sieci neuronowej z parametryzowanymi konkluzjami reguł.

Zastosowany w systemie podział hierarchiczny polega na wyborze najgorszego regionu (tzn. takiego, który ma największy przyczynek do końcowego błędu systemu), następnie region ten jest dzielony. Po podziale i uzyskaniu w miejsce jednego regionu – dwóch regionów potomnych, wszystkie regiony są strojone metodą gradientową. Pozwala to na lepsze ich dopasowanie do danych i wypracowanie postaci konkluzji.

W artykule przedstawiono także wyniki badań eksperymentalnych dla zbiorów zarówno syntetycznych, jak i zbiorów zawierających wyniki rzeczywistych pomiarów. Wyniki tych badań skłaniają do wyciągnięcia wniosku, że podział hierarchiczny dziedziny umożliwia uzyskanie systemu dokładniejszego, jeśli chodzi o generalizację wiedzy.

Address

Krzysztof SIMIŃSKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16,
44-100 Gliwice, Polska, krzysztof.siminski@polsl.pl