Feature Description Systems for Clusters by Using Logical Rule Generations Based on the Genetic Programming and Its Applications to Data Mining

Jianjun Lu[1,2], Yunling Liu[1], and Shozo Tokinaga[2]

[1] China Agricultural University, Beijing 100083, China
[2] Graduate School of Economics, Kyushu University, 812-8581 Japan

Abstract. This paper deals with the realization of retrieval and feature description systems for clusters by using logical rule generations based on the Genetic Programming (GP). At first, whole data is divided into several clusters and the rules are improved based the GP. The fitness of individuals is defined in proportion to the hits of corresponding logical expression to the samples in targeted cluster c, but also in inversely proportion to the hits outside the cluster c. The GP method is applied to various real world data by showing effective performance compared to conventional methods.

1 Introduction

This paper deals with the feature description systems for clusters by using rule generations based on the Genetic Programming (GP) and its applications to data mining [4][5]. In the method, we prepare various kinds of logical expression (having tree structure and are called as individuals) for the data (called as samples in the following) in the underlying cluster by using variables for categorical values, then we improve the logical expressions by using the GP. As the fitness of each individual, we use the number of hits of individual (cases where the logical expression corresponding the individual) for the samples in cluster[1]-[3].

As applications, we apply the method to the evaluation of decision making of personal loan by showing the effectiveness of the method. Moreover, the feature description system is applied to eight groups of samples which are arbitrarily collected from various data bases.

2 Feature Description System for Cluster Based on the GP

The overview of the feature description system for cluster based on the GP treated in the paper is shown in the following [4].
(1) Description of samples by categorical variables

Y. Shi et al. (Eds.): ICCS 2007, Part IV, LNCS 4490, pp. 162–165, 2007.

Generally, there are two kind of variables (numerical variables and logical variables) to characterize the sample, but we assume that in the system we have only logical variables. The numerical variables are transformed into categorical variables by using conventional methods of discretization (details are omitted here)[6].

(2) generation of initial individual for logical expressions

It is assumed that the logical expressions represented by the categorical variables are used to describe the feature of samples in the cluster which correspond to the individuals. For the sake of tractability, it is assumed that the logical expression has a binary tree structure. At the beginning of GP procedure, we generate the initial pool of individuals (say, 1000 individuals) by using random numbers.

(3) definition of fitness of individuals

The definition of the fitness of k th individual in the GP procedure is given by the number of hits denoting the number of samples in the cluster to which the logical expression corresponding to the individual is true. Moreover, the number of hits for the samples outside the underlying cluster c in the whole dataset is also used. We at first defines following index.

$$y_k = T - h_k^2/n_k \tag{1}$$

The notations included in equation (1) are defined as follows.

n_k : number of samples in the whole dataset for which the logical expression of the individual k is true

h_k : number of samples in the underlying cluster c for which the logical expression of the individual k is true

T : total number of samples in the cluster c

The fitness of individual k (denoted as f_k) is defined by adding a certain positive number to y_k, and taking the inverse of the number as.

$$f_k = (a + y_k)^{-1} \tag{2}$$

If the number of hits for logical expression covering samples in the cluster becomes larger, then accordingly, the measure defined in equation (1) becomes to be close to zero.

3 Logical Rules for Feature Description and the GP

The prefix representation is equivalent to the tree representation of arithmetic expressions[6][7]. For example, we have the next prefix representation.

$$(6.43x_1 + x_2)(x_3 - 3.54) \longrightarrow \times + \times 6.43x_1x_2 - x_3 3.54 \tag{3}$$

We apply the GP procedure to approximate the functions f(,) using the observations y_n.

To keep the consistency of genetic operations, the so-called stack count (denoted as *StcckCount*) is useful. The *StackCount* is the number of arguments it

places on minus the number of arguments it takes off the stack. The cumulative *StackCount* never becomes positive until we reach the end at which point the overall sum still needs to be 1. The basic rule is that any two loci on the two parents genomes can serve as crossover points as long as the ongoing *StackCount* just before those points is the same. The crossover operation creates new offsprings by exchanging sub-trees between two parents.

Usually, we calculate the root mean square error ($rmse$) between $x(t)$ and $\tilde{x}(t)$, and use it as the fitness. The fitness S_i of i th individual is defined as the inversion of $rmse$.

We assume that the all of samples are characterized by the logical variables, then we use a relatively simple method to generate logical expressions. Assume that there are categorical variables $v_1, v_2, ..., v_m$, and these variables can have the values $s_1, s_2, ...$ For example, if the logical variables v_1, v_2 take the value s_3, s_5, then we have

$$v_1 = s_3, v_2 = s_5 \tag{4}$$

Then, these binary expressions are used as predicates included in the logical expressions in the GP. For example, we define new logical variables X_{kj} represented by input variable v_i such as

$$X_{kj} = \begin{cases} True, & \text{if } v_k = s_j \\ False, & \text{otherwise} \end{cases} \tag{5}$$

We also define the fitness of individuals as the accuracy of rule generated by the rule corresponds to the underlying individual. To improve the fitness of individuals, we apply the GP operations to the logical expressions.

4 Applications

4.1 Applications to German Credit Data

The experiment on real-life credit-risk evaluation is carried out using the German credit data. The German credit data is obtainable from the website. The data consist of 1000 records of personal loan, and the input variables for one record include 7 numerical data and 13 categorical data.

Even though the original purpose of the dataset is the generation of accept/deny rules for personal loans, but we use the dataset to examine the ability of the GP method of the paper. At first, we select 100 samples at random from the dataset, and classify them into three clusters by using following seven numerical variables based on the conventional software package.

Then, we assume one cluster (say cluster c) is the target cluster whose feature must be described, and other 6 samples belonging two another clusters are regarded as samples outside cluster c.

It is concluded that after about 500 generation of the GP procedures the feature extraction (description) is completed, and the logical expression finally obtained describes the true feature of cluster c.

4.2 Applications of Feature Description to Real Data

In the following, we explain the simulation studies of the feature description of the paper applied to multiple real dataset, and discuss the performance in the mean.

At first, we select about 100-300 samples from the dataset at random, and then we divide these samples into three cluster by using conventional numerical method of clustering. Then, at the next step, we apply the GP procedure of the paper for the extraction of cluster and feature description to these three clusters, independently. That means if we focus on the cluster c, the samples belonging two clusters d different from the underlying cluster c are regarded as the samples outside the cluster c. The number of GP generation N_F necessary to obtain final result of feature description is omitted. The GP procedure to extract the clusters and to give feature descriptions can work effectively by spending 500 or 600 GP generations even for real world data, despite wide variations. We must note that by using the feature description method of the paper, finally we obtain 100% correct classification of samples to the underlying clusters.

5 Conclusion

This paper treated the realization of retrieval and feature description systems for clusters by using logical rule generations based on the GP. As applications, the GP method was applied to various real world data. For future works, it is necessary to apply the method of transformation of logical expressions to natural language. Further works by the authors will be continued.

References

1. G.Piatetsky and W.J.Frawley,"Knowledge discovery in database:An overview,"in Knowledge Discovery in Database, AIII/MIT Press, 1991.
2. A.A.Freitas, Data Mining and Knowledge Discovery with Evolutionary Algorithms, Springer-Verlag, 2002.
3. S.Tokinaga,J.Lu and Y.Ikeda, "Neural network rule extraction by using the Genetic Programming and its applications to explanatory classifications," IEICE Trans.Fuadamentals, vol.E88-A,no.10,pp.2627-2635,2005.
4. M.L.Wong and K.S.Leung,Data Mining Using Grammer Based Genetic Programmign and Applications,Kluwer Academic Publisher, London,2000.
5. J.Lu, Y.Kishikawa and S.Tokinaga,"Realization of Feature Descriptive Systems for Clusters by using Rule Generations based on the Genetic Programming and its Applications(in Japanese)," IEICE Trans.Fuadamentals, vol.J89-A,no.12,pp.2627-2635,2006.
6. J.Lu, S.Tokinaga, and Y.Ikeda, gExplanatory Rule Extraction Based on the Trained Neural Network and the Genetic Programmingh, Journal of the Operations Research Society of Japan, Vol.149, No.1, pp.66-82, 2006.
7. Y.Ikeda and S.Tokinaga,"Chaoticity and fractality analysis of an artificial stock market by the multi-agent systems based on the co-evulutionary Genetic Programming", IEICE Trans.Fundamentals, vol.E87-A,no.9,pp.2387-2394, 2004.
8. J.R.Koza,Genetic Programming,MIT Press, 1992.