STUDIA UNIV. BABEŞ-BOLYAI, INFORMATICA, VOLUME XLIII, NUMBER 2, 1998

HIERACHICAL DATA STRUCTURE DETECTION USING EVOLUTIONARY ALGORITHMS

DAN DUMITRESCU LAVINIA HUI
BEATRICE LAZZERINI

Abstract. A new evolutionary algorithm for detecting the hierarchical structure of a data set is proposed. The method is particularly suitable for clustering purpose. In this case the considered approach may also supply the optimal cluster number in the data set.

1. Introduction

This paper proposes a method to detect the hierarchical structure of a data set using a genetic algorithm (see [1,2])

Let us consider a data set $X=\{x^1,x^2,...,x^p\}$, and let d be a distance on X. Our aim is to detect hierarchical cluster structure of X. The problem is particularly difficult when the optimal cluster number n is unknown ([3]). The method proposed in this paper will give the optimal number of classes as well as their hierarchical organisation.

For this purpose a genetic algorithm approach will be used. Each chromosome will describe a cluster hierarchy. In order to obtain a feasible solution, we need a chromosome representation, which ensures that the search space would be completely explored. This means that all possible solutions may be generated.

2. Representing a hierarchy

To describe a hierarchy we have to follow some steps:

a) First we will consider a binary tree having 2^{p-2} nodes. This nodes will be labelled by numbers from 1 to 2^{p-2}, starting with the root node (level zero). The i-th level, $0 \le i \le (p-3)$ contains 2^i nodes. Taking the sum of the nodes from the first $(p-3)$ levels we have:

$$\sum_{i=0}^{p-3} 2^i = 2^{p-2} - 1$$

From this equality we deduce that the first $(2^{p-2}-1)$ nodes are distributed in the first $(p-3)$ levels of the tree. Therefore the level $(p-2)$ will contain only one node.

The considered tree represents a skeleton of a hierarchical structure. The nodes in this tree will be non-terminal nodes in the tree describing the classification hierarchy. For this reason we call them non-terminal or skeletal nodes.

b) After we prepare this skeleton tree we consider a string having p positions. Each position contains an integer number c_j, $1 \le c_j \le 2^{p-2}$. Each position indicates a link

between a data points x^j and a skeletal node. In our case because we use a genetic algorithm, this string is represented by a chromosome with a real codification.
In the chromosome:

$$c = (c_1, c_2, \ldots, c_p)$$

the value c_j of the gene j indicates the non-terminal node to which the point x^j (represented by a new terminal node) is attached. So, the data point that we want to classify will be represented by nodes that "hang" in the skeleton tree. These will be our terminal nodes.

c) To obtain a data hierarchy, after all attachments are performed we have to do some transformations. We have to check bottom-up all nodes of the structure. If a skeleton node has maximum one descendant, the skeleton node is removed from the structure and its son is connected to the upper node (see also [4]).

EXAMPLE:

We want to represent a space who has five data and the classes of these data are:

$$\{x^1, x^4\},$$
$$\{\{x^2, x^3\}, x^5\}.$$

We will use a tree with $2^{5-2} = 8$ skeletal nodes. Let us now the chromosome c = (3 5 4 5 2). The skeletal tree representing this chromosome is depicted in Figure1.

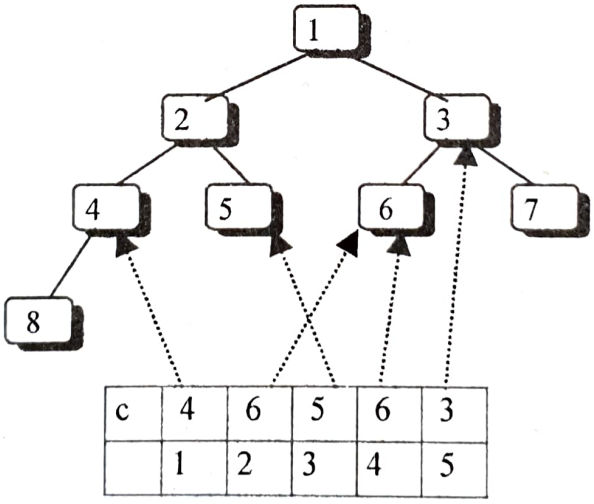


Figure 1. A skeletal tree for five data points

The corresponding decision tree is depicted in Figure2.

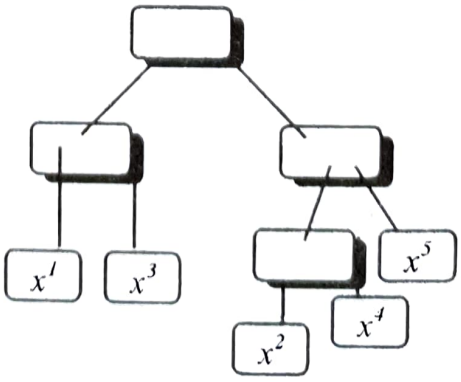


Figure 2. The decision tree corresponding to the chromosome c = (4 6 5 6 3)

The proposed mechanism may generate a new tree which describes a hierarchy cluster structure on X. This tree has two kind of nodes. The skeleton nodes are non-terminal ones, and the data points are represented by terminal nodes. In this new tree, if two terminal nodes are connected with the same node we consider the data points represented by these nodes belongs to the same cluster. So, each non-terminal node represents a class in X.

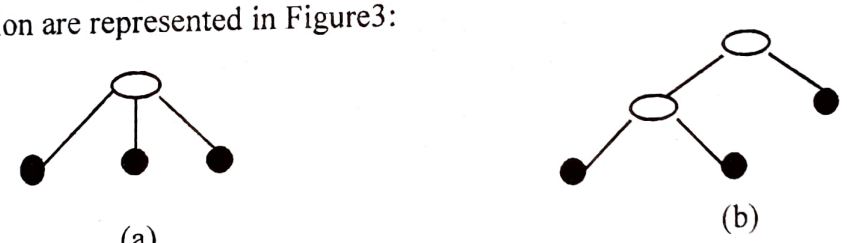
The proposed approach allows us to obtain the optimal cluster number n. The number is given by the number of the non-terminal nodes.

This mechanism may generate a non-binary tree. The following theorem ensures that this method can generated all hierarchical structures corresponding to p points.

Theorem 2.1 Any hierarchy of p objects may be described by using the previous method.

Proof. It is done by induction with respect to p. The steps of the proof are given below.

1. If we have a data set with three objects (p=3) we can have two situations. This situation are represented in Figure3:



(a) (b)

Figure 3. The hierarchical structures for three data points

We have to prove that these hierarchies are obtained by our method. According to this method we have $2^{3-2}=2$ nodes in the skeleton tree. The corresponding skeleton tree is represented in Figure4.

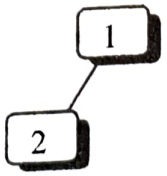


Figure 4. The skeleton node for three data points

If we attach to the non-terminal node labelled 2 two terminal nodes and to the root node one terminal we obtain the structure from the Figure 3(b). This hierarchical structure is described by the chromosome c_1=(2 2 1).

Let us consider that all terminal modes are linked to the root. In this case the skeleton mode labelled 2 is deleted (it has not an offspring). We have obtained the structure represented in Figure 3(a). The corresponding chromosome is c_2=(1 1 1).

2. Let us consider k data points. We prepare a tree structure with 2^{k-2} nodes and a chromosome who represent a hierarchy. We suppose that a chromosome can represent any hierarchy induced by k points.

3. Let us now consider $(k+1)$ objects. This means that we add one new object to the tree considered at the step 2. From the data set point of view this point can be added in 3 different kinds:

a) the point belong to a class (i)
b) the point does not belong to any class
c) the point induce a new class in our hierarchy

The first two situations are treated in the same way. We use the hierarchy establish to the step 2. We add one more terminal node to the non-terminal node labelled i in the first case, or to the root node in the second situation.

In the third situation the input space will have a new class so the tree structure will have one more non-terminal node. The problem is if the skeleton node has now enough nodes for representing this structure. The new skeleton tree will have with $2^{k-1}-2^{k-2}=2^{k-2}$ more nodes as the skeleton tree prepared in step 2. So, in the k-skeleton tree we will complete the whole last level, and we will add a new level with one node only. The number of the data classes is increased only with one, so the number of levels in the hierarchy tree may also increase only with one level. But as we see this one more level is prepared even from the skeleton tree.

3. Coding

Each chromosome with p genes will describe a cluster hierarchy on X. To each hierarchy that we consider corresponded a number n of virtual clusters. The number of the non-terminal nodes gives this number, each non-terminal node corresponds to a virtual class A_i.

The prototype of a virtual class A_i is a new data point L^i who depends on all data point for class A_i. Virtual classes may contain data points as well as prototypes.

We may admit that two terminal nodes hanging the same parent belong to the same real class. Within this assumption the real classes may be obtained from the virtual classes by deleting the prototypes. Let B_i be the real class induced by the virtual class A_i. We may formally write:

$$B_i = A_i \setminus \{ L^k \mid L^k \in A_i \}$$

Example 3.1 Let us consider the hierarchical structure described in Figure 2, and the chromosome c = (4 6 5 6 3). The five points for our data set determine four virtual classes. These classes are:

$A_4=\{x^2,x^4\}$ and the prototype for this class is L^4,
$A_3=\{L^4,x^5\}$ and the prototype for this class is L^3,
$A_2=\{x^1,x^3\}$ and the prototype for this class is L^2,

$A_1=\{L^2,L^3\}$ and the prototype for this class is L^1.

The hierarchy can be expressed now as indicated in Figure 5.

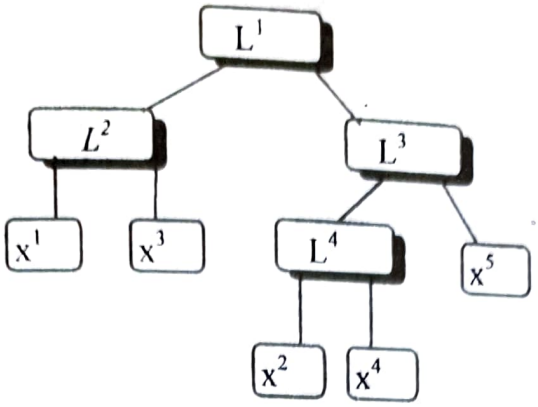


Figure 5. A data hierarchy for five points

The class A_1,A_2,A_3,A_4 corresponding to the non-terminal nodes do not describe the cluster structure. The cluster structure is represented by the partition B_2,B_3,B_4 of X, where:

$B_4=\{x^2,x^4\}$,

$B_3=\{x^5\}$,

$B_2=\{x^1,x^3\}$.

4. Fitness function

Let us consider a data set $X=\{x^1,x^2,...,x^p\}$, Our aim is to detect the hierarchical cluster structure of X, so we have to find a chromosome who represents the best cluster structure of X. For this purpose we will use a genetic algorithm.

For evaluating each chromosome we need a fitness function to measure the quality of hierarchy that we obtain. This function must have the best value for the hierarchy that is the most suitable for the data set.

We propose to use the following fitness function:

$$f(c) = Max - \sum_{i=1}^{m} \sum_{x^k \in A_i} d(L^i, x^k),$$

where:

m is the virtual cluster number,

L^i is the prototype of the virtual class A_i,

Max is a value such that $f(c) \geq 0, \forall c$.

The prototype L^i is the mean vector of the virtual class A_i:

$$L^i = \frac{\sum_{x \in A_i} x}{p_i},$$

where p_i is the cardinality of A_i.

We give two examples to illustrate how the values of this fitness function change for different hierarchy cluster structure.

It is more convenient to consider the fitness function to be minimised is:

$$g(c) = \sum_{i=1}^{m} \sum_{x^k \in A_i} d(L^i, x^k).$$

Example.

We consider the data set: $X=\{x^1,x^2,x^3,x^4\}$, $X \subset \boldsymbol{R}^2$.

a) Consider the data points as depicted in Figure 6: $x^1=(1,1)$, $x^2=(2,1)$, $x^3=(2,3)$, $x^4=(5,4)$.

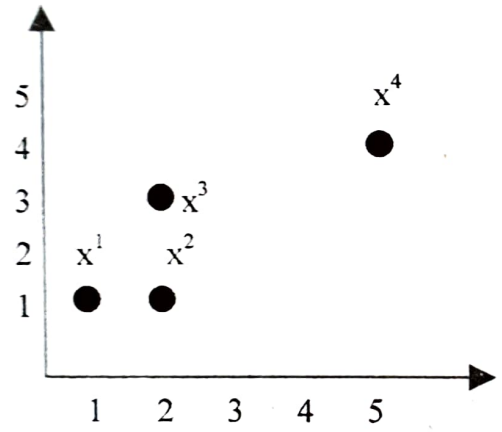


Figure 6. The data points distribution

Within this data set we can establish different hierarchies. We will consider those we consider are more important. We will describe the hierarchies by expressing the virtual classes.

Case 1. Consider the chromosome c_1=(2 2 3 3). The virtual classes and the corresponding prototype are:

$A_2=\{x^1,x^2\}$,

$L^2=(3/2,1)$,

$A_3=\{x^3,x^4\}$,

$L^3=(7/2,7/2)$,

$A_1=\{L^2,L^3\}$.

The value of the fitness function g is:

$$g(c_1) = 1 + \sqrt{10} + \frac{\sqrt{41}}{2} \approx 7.3638.$$

Case 2. Consider the chromosome c_2=(4 4 2 1). The virtual classes and the corresponding prototype are:

$A_3=\{x^1,x^2\}$,
$L^3=(3/2,1)$,
$A_2=\{x^3,L^3\}$,
$L^2=(7/4,2)$,
$A_1=\{x^4,L^2\}$.

The value of the fitness function g is:

$$g(c_2) = 1 + \frac{\sqrt{17}}{2} + \frac{\sqrt{233}}{4} \approx 6.8776.$$

Case 3. Consider the chromosome c_3=(1 1 1 1). The chromosome describes one virtual class: $A_1=\{x^1,x^2,x^3,x^4\}$. The class prototype is $L^1=(10/4,9/4)$. The value of the fitness function g is:

$$g(c_3) = \frac{\sqrt{61}}{4} + \frac{\sqrt{29}}{4} + \frac{\sqrt{13}}{4} + \frac{\sqrt{149}}{4} \approx 7.2518.$$

Case 4. Consider the chromosome c_4=(2 2 1 1). The virtual classes and the corresponding prototype are:

$A_2=\{x^1,x^2\}$,
$L^2=(3/2,1)$,
$A_1=\{x^3,x^4,L^2\}$,
$L^1=(17/6,8/3)$.

The value of the fitness function g is:

$$g(c_4) = 1 + \frac{\sqrt{29}}{6} + \frac{\sqrt{233}}{6} + \frac{\sqrt{164}}{6} \approx 6.5759.$$

Case 5. Consider the chromosome c_5=(2 2 2 1). The virtual classes and the corresponding prototype are:

$A_2=\{x^1,x^2,x^3\}$,
$L^2=(5/3,5/3)$,
$A_1=\{x^4,L^2\}$.

The value of the fitness function g is:

$$g(c_5) = \frac{\sqrt{8}}{3} + \frac{\sqrt{5}}{3} + \frac{\sqrt{17}}{3} + \frac{\sqrt{149}}{3} \approx 7.1313.$$

We notice that from the five different situations that we analyse the less value for the fitness function corresponds to case 4. That means that this is the most representative hierarchy for our input space. The hierarchy is depicted in Figure 7.

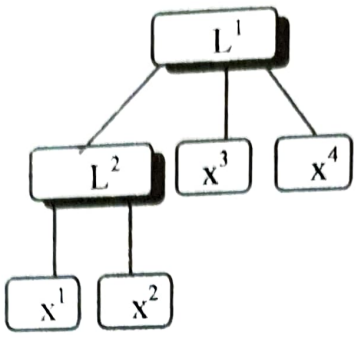


Figure 7. The best hierarchy structure for the data set represent in fig6, for c=(2 2 1 1)

The value of the fitness function corresponding to the structure described in Figure 7 for the chromosome c=(2 2 1 1) may be written as:

$$f(c) = Max - [d(x^1,L^1) + d(x^2,L^1) + d(x^3,L^2) + d(x^4,L^2) + d(L^1,L^2)].$$

The corresponding real classes are:

$B_2=\{x^1,x^2\}$,

$B_1=\{x^3,x^4\}$.

b) Consider another example where the four points of the data set are placed in the corner of a square, as depicted in Figure 8.

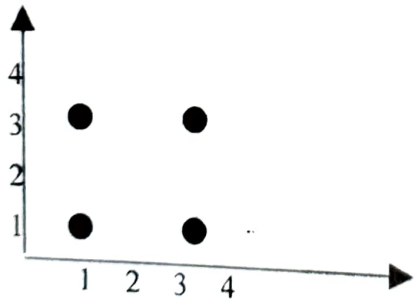


Figure 8. The data points distribution

$x^1=(1,1)$, $x^2=(3,1)$, $x^3=(3,3)$, $x^4=(1,3)$.

Case1. Consider the chromosome c_1=(2 2 3 3). The virtual classes and the corresponding prototype are:

$A_2=\{x^1,x^2\}$,

$L^2=(2,1)$,

$A_3=\{x^3,x^4\}$,

$L^3=(2,3)$,

$A_1=\{L^2,L^3\}$.

The value of the fitness function g is:

$$g(c_1)=2+2+2=6.00\ .$$

Case 2. Consider the chromosome c_2=(2 2 2 1). The virtual classes and the corresponding prototype are:

$A_2=\{x^1,x^2,x^3\}$,
$L^2=(7/3,5/3)$,
$A_1=\{x^4,L^2\}$.

The value of the fitness function g is:

$$g(c_2)=\frac{\sqrt{20}}{3}+\frac{2\sqrt{2}}{3}+\frac{\sqrt{20}}{3}+\frac{4\sqrt{2}}{3}\approx 5.80.$$

Case 3. Consider the chromosome c_3=(1 1 1 1). The chromosome describes one virtual class: $A_1=\{x^1,x^2,x^3,x^4\}$. The class prototype is $L^1=(2,2)$. The value of the fitness function g is $g(c_3)=4\sqrt{2}\approx 5.65$.

Case 4. Consider the chromosome c_4=(4 4 2 1). The virtual classes and the corresponding prototype are:

$A_3=\{x^1,x^2\}$,
$L^3=(2,1)$,
$A_2=\{x^3,L^3\}$,
$L^2=(5/2,2)$,
$A_1=\{x^4,L^2\}$.

The value of the fitness function g is:

$$g(c_4)=2+\sqrt{5}+\frac{\sqrt{13}}{2}\approx 6.03\ .$$

As we expected the best value are obtained in the case number 3. There is not any preference between these four data points (they are equally distributed). The hierarchy is depicted in Figure 9.

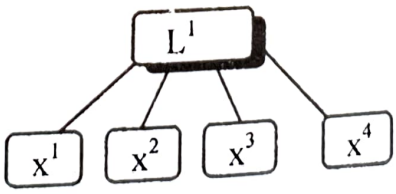


Figure 9. The best hierarchy structure for the data set represented in fig. 8, for c=(1 1 1 1)

The value of the fitness function corresponding to the structure described in Figure 9 for the chromosome c=(1 1 1 1) may be written as:

$$f(c)=Max-[d(x^1,L^1)+d(x^2,L^1)+d(x^3,L^1)+d(x^4,L^1)].$$

The corresponding real class is $B_1=\{x^1,x^2,x^3,x^4\}$.

5. The Genetic Algorithm

A genetic algorithm with crossover and mutation may be used to obtain the best hierarchical cluster structure of a data set.

By mutation a gene is replaced with a symbol randomly chosen from the set $1,2,\ldots,2^{n-2}$.

For recombination the uniform crossover operator ([1]) may be used. By uniform crossover the gene of a descendent is selected from any parent with a given probability.

REFERENCES

[1]. Bock T., Fogel D.B., Michalewicz, Z, *Handbook of Evolutionary Computation*, Oxford University Press, Oxford, 1997.

[2]. Dumitrescu, D., Lazzerini, B., Jain, L., Dumitrescu, A., *Evolutionary Computation*, C.R.C.,1999.

[3]. Dumitrescu , D., Stan , I., Dumitrescu , A., *Genetic algorithms in fuzzy clustering*, EUFIT 1997, Aachem 705-708.

[4]. Skimojima , K., Fukunda ,T., Hasegawa ,Y., *Self-turning fuzzy modelling with adaptive membership function, rules, and hierarchical structure based on genetic algorithm*, Fuzzy sets Systems, 7(1995), 295-309.

Babeş-Bolyai University, Faculty of Mathematics and Informatics, RO 3400 Cluj-Napoca, str. M. Kogalniceanu 1, România.

E-mail address: ddumitr@cs.ubbcluj.ro

Dipartimento di Ingegneria della Informazione, Universita di Pisa, Italy

E-mail address: beatrice@iet.unipi.it