A GENETIC ALGORITHM FOR MULTICRITERIA INVENTORY CLASSIFICATION

H. Altay Güvenir

Department of Computer Engineering and Information Science,
Bilkent University, Bilkent, 06533 Ankara

guvenir@cs.bilkent.edu.tr

In *Artificial Neural Nets and Genetic Algorithms, Proceedings of the International Conference* Ales, France, April 1995, D. W. Pearson, N. C. Steele, and R. F. Albrecht (Eds), Springer-Verlag, Wien, (pp. 6-9).

BU-CEIS-9504

A GENETIC ALGORITHM FOR MULTICRITERIA INVENTORY CLASSIFICATION

H. Altay Güvenir

Department of Computer Engineering and Information Science,
Bilkent University, Bilkent, 06533 Ankara
guvenir@cs.bilkent.edu.tr

Abstract

One of application areas of the genetic algorithms is parameter optimization. This paper addresses the problem of optimizing a set of parameters that represent the weights of criteria, where the sum of all weights is 1. A chromosome represents the values of the weights, possibly along with some cut-off points. A new crossover operation, called *continuous uniform crossover*, is proposed, such that it produces valid chromosomes given that the parent chromosomes are valid. The new crossover technique is applied to the problem of multicriteria inventory classification. The results are compared with the classical inventory classification technique using Analytical Hierarchy Process.

INTRODUCTION

Management and control of inventories consisting of a large number of items is usually done by classifying these items in three groups, called the *ABC classification*. The items that require closest attention are called class A items. The class C items are given the least importance. All other items are classified as class B. In the traditional ABC classification, items are classified according to their total annual dollar usage. It has been suggested by Flores and Whybark (1986) that multiple criteria ABC classification can provide a more comprehensive managerial approach. Flores et. al. (1992) proposed the use of the Analytic Hierarchy Process (AHP) to reduce these multiple criteria to a univariate and consistent measure to consider multiple inventory management objectives. Through subjective pairwise comparisons of the criteria, a weight vector is derived. The elements of this vector represent the absolute weights of each criteria. The inventory items are then sorted according to this weight vector. The items on top of this sorted list are classified as class A, and the ones in the bottom (usually lower half) as class C. The remaining items are classified as class B. The difficulty with this approach is that it is quite difficult to reflect the relative weight of criteria in the form of pairwise comparisons. To remedy this problem we propose a method, which uses direct classifications of a small set of training items. The method we propose here uses a genetic algorithm to learn a weight vector, along with the cut-off points between class A and B, and class B and C. The remaining set of items are then classified using this weight vector and cut-off points. Note that the approach proposed here is applicable to any multicriteria classification problem with any number of classes, provided that it is possible to reduce the problem to learning a weight vector along with the cut-off points between classes.

We compared the pairwise comparison (AHP) technique and the genetic algorithm approach on a sample inventory. We have noticed that are much more positive in classifying items than subjectively comparing two criteria. The classification made by our technique was closer to the desired classification than the one made by the AHP technique.

An important characteristic of the weight vector is that the elements are real values representing absolute weights of the criteria whose sum is 1. In order apply a genetic algorithm to the weight learning problem, we proposed a new crossover operator that guarantees the generation of off-spring that are valid representations of weight vectors. This approach is better than using the standard crossover operations and controlling the search through penalty functions only.

In the rest of the paper we describe the AHP technique, then define our approach. Next we compare both techniques on a sample inventory. The paper concludes with general remarks on the use of genetic algorithms in multicriteria classification.

MULTICRITERIA INVENTORY CLASSIFICATION USING AHP

AHP is proposed by Saaty (1980) as a general theory of measurement. Application of AHP to multiple criteria classification requires a pairwise comparison of all criteria by an expert. The decision maker is usually asked to assign a value in a scale 1 through 9 for each pair of criteria. These pairwise comparisons are combined in a square matrix called *pairwise comparison matrix*. Given a pairwise comparison matrix A, the entry A_{ij} represents the relative strength of criterion C_i when compared to C_j. That is, A_{ij} is expected to be w_i/wj, where w_i and wj are the relative strengths of the criteria C_i and C_j, respectively. Therefore, the pairwise comparison matrix is a symmetric matrix and its diagonal is equal to 1. If the expert was not consistent in the pairwise comparisons the resulting matrix will be inconsistent. In that case, the expert is asked to form the pairwise comparison matrix again. This process repeated until the matrix is consistent. The next step consists of the computation of a vector of priorities from the constructed matrix. According to the AHP methodology, the eigenvector of the comparison matrix with the largest eigenvalue provides the priority ordering of the criteria, that is their weights. The values of this vector are between 0 and 1, and their sum is 1.

For each inventory item, the values of each criteria are, then, organized in a way that the higher values increase the probability of an item being in Class A. The criteria values are normalized to the range of 0 to 1 for each criteria. The weighted score of each item in the inventory is computed as the sum of the product of weight and normalized value for each criteria. The items are sorted in descending order according to their weighted score values. Finally, the first 20% of the items are classified as class A, the next 20% as class B, and the remaining items as class C.

Although this technique is sound, the pairwise comparisons are based on the subjective judgments of an expert. Usually, the expert does not agree with the resulting classification of the inventory. In that case, the expert changes the pairwise comparisons and obtains a new weight vector. This process continues until there are no obvious misclassification of the inventory items. This difficulty arises because it is not easy to compare two criteria and assign a numerical value. However, experts are able to judge the appropriateness of the resulting classification.

MULTICRITERIA INVENTORY CLASSIFICATION USING GA

Here we propose an alternative method to learn the weight vector along with the cut-off values for multi-criteria inventory classification. The method proposed here, called GAMIC (for Genetic Algorithm for Multi-criteria Inventory Classification), uses a genetic algorithm to learn the weights of criteria along with AB and BC cut-off points from preclassified items. Once the criteria weights are obtained the weighted scores of the items in the inventory are computed as in the previous method. Then the items with score greater than AB cut-off value are classified as A, those with values between AB and BC as class B, and the remaining items are classified as class C.

Encoding

A chromosome encodes the weight vector along with two cut-off points. The values of the genes are real values between 0 and 1. The total value of the elements of the weight vector is always 1. Also the AB cut-off value (x_{AB}) is always greater than the BC cut-off value (x_{BC}). Therefore, if the classification is based on k criteria, a chromosome $\mathbf{c}$ is a vector defined as

$$\mathbf{c} = < w_1, w_2, \ldots w_k, x_{AB}, x_{BC} > .$$

Here, w_j represents the weight of jth criterion, $\sum_{j=1}^{k} w_j = 1$, and $x_{AB} < x_{BC}$. In this representation the relative weight of a criteria can be encoded as an absolute value in a chromosome. That is, an allele represents the weight of the corresponding criterion, independent of the other alleles.

Given a chromosome $\mathbf{c}$, classification of an inventory item i is done by computing its weighted sum, $ws_{c,i}$ as follows:

$$ws_{c,i} = \sum_{j=1}^{k} w_j \frac{i_j - min_j}{max_j - min_j}.$$

Here i_j is the value of the item i for the criterion j, max_j and min_j are maximum and minimum values of criterion j among all inventory items. The classification of an inventory item i according to chromosome $\mathbf{c}$ is

$$classification(i, \mathbf{c}) = \begin{cases} A & \text{if } ws_{\mathbf{c},i} \leq x_{AB} \\ B & \text{if } x_{BC} \leq ws_{\mathbf{c},i} < x_{AB} \\ C & \text{otherwise} \end{cases}$$

Given this encoding scheme, GAMIC applies the standard genetic operators (reproduction, crossover, and mutation) to the chromosomes in the population (Goldberg, 1989). GAMIC applies fitness proportionate roulette wheel selection in reproduction. GAMIC also uses the elitist approach, that is the best chromosome is always copied to the next generation. The evaluation of the fitness of a chromosome, the crossover, and the mutation operations employed by GAMIC are described below.

Fitness function: The fitness of a chromosome reflects its ability to classify the training set correctly. Therefore, any misclassified item should introduce a penalty. However, due the the linear ordering among the classes, we have to distinguish the error made by classifying an A as B than as C. In our implementation the fitness of a chromosome $\mathbf{c}$ was computed as

$$fitness(\mathbf{c}) = \frac{\sum_i^t p_i}{t},$$

$$p_i = \begin{cases} 1 & classifcation(i,c) = class(i), \\ 0.4 & |classifcation(i,c) - class(i)| = 1, \\ 0 & otherwise, \end{cases}$$

where t is the size of the training set. Note that this fitness function prefers a chromosome making two mistakes with difference 1 to a chromosome making a single mistake with difference of 2.

Crossover

Although the initial population is setup in a way that all chromosomes represent legal codings (each allele is between 0 and 1, the sum of all weight values is 1, and the AB cut-off is less than the BC cut-off), standard crossover operations are bound to result in illegal codings. Here we propose a new form of uniform crossover operation, called *continuous uniform crossover*, that guarantees the legality of the offsprings.

Continuous Uniform Crossover: Given two chromosomes $\mathbf{x} = < x_1, x_2, \ldots, x_n >$, and $\mathbf{y} = < y_1, y_2, \ldots, y_n >$, the offsprings are defined as $\mathbf{x}' = < x'_1, x'_2, \ldots, x'_n >$, and $\mathbf{y}' = < y'_1, y'_2, \ldots, y'_n >$, where

$$\begin{aligned} x'_i &= sx_i + (1-s)y_i \\ y'_i &= (1-s)x_i + sy_i \end{aligned}$$

Here s, called *stride*, is constant through a single crossover operations. This crossover preserves the sum of any subset of genes. In the case of inventory classification, if the genes 1 through m encode the criteria weights, $\sum_{i=1}^m x_i = 1$. After the crossover operation,

$$\sum_{i=1}^m x'_i = s\sum_{i=1}^m x_i + (1-s)\sum_{i=1}^m y_i = s + (1-s) = 1.$$

This is true for both offsprings; $\sum_{i=1}^m x'_i = 1$, and $\sum_{i=1}^m y'_i = 1$, Further, this crossover preserves the greater-than relation between genes, as well. That is, if $x_{AB} > x_{BC}$, then $x'_{AB} > x'_{BC}$. Therefore, continuous uniform crossover preserves the legality of chromosomes for the multicriteria ABC classification.

The choice of the stride (s) is an important issue. If $s = 0$, the off-springs are the same as the parents. If $0 \leq s$ then alleles remain to be between 0 and 1, however the alleles get closer to each other through generations. If $s = 0.5$, then both off-springs are the same, and the alleles are equal to the average values of the respective alleles in the parents. On the other hand, if $s < 0$, the alleles diverge from the respective values in the parents. However, if $s < 0$, then alleles may be outside of the limits; that is an allele may get a negative value of a value greater than 1, although their sum is still 1. In that case, a normalization of the chromosome is needed. In our implementation we chose s randomly from [-0.5, 0.5] for each crossover operation.

Mutation: Mutation operation in our implementation sets the value of a gene to 0 or 1 with equal probability. If a chromosome is modified by the mutation operation, it has to be normalized.

EXPERIMENT

We have compared these two techniques (AHP and GAMIC) for multicriteria classification on an inventory of 115 items used by a construction company in rock excavation jobs. The items are to be classified according to ten criteria. We first asked an expert to name five items from each of the three classes. This set of 15 items formed our training set for the GA. The genetic algorithm was run on a population of 100 chromosomes, with $p_c = 0.7$ and $p_m = 0.001$. Our genetic algorithm converged on the training set in the 340th generation after 24003 function evaluations. The best and average fitness values in each generations are shown in Figure ??.

We classified the remaining 100 items using the weights and cut-off values learned by the GA. The expert also was asked to classify these test items. The expert's

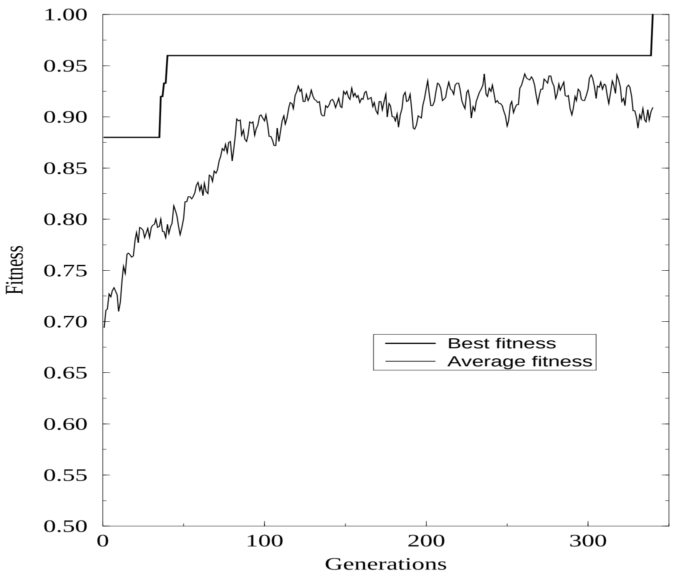

Figure 1: Best and average fitness values through generations.

classification did not agree with GA method on 7 items. In order to compare with the AHP technique, the expert was asked to pairwise compare the criteria. In the second try the expert obtained a consistent matrix. All 115 items were then classified according to the weights learned by the AHP technique. The expert did not agree with the classification of 13 items. As a result the genetic algorithm method proposed here performed better than the AHP technique in our test set.

CONCLUSION

We presented a new approach using genetic algorithms to multicriteria classification. This approach is based on learning a weight vector of absolute weights of each criteria along with a set of cut-off points. In order apply a genetic algorithm to the weight learning problem, we proposed a new crossover operator that guarantees the generation of off-spring that are valid representations of weight vectors.

We have compared our approach with classical AHP technique based on subjective pairwise comparisons of criteria on a sample inventory classification task. The approach, implemented in a program called GAMIC, is applicable to any multicriteria classification problem with any number of classes, provided that it is possible to reduce the problem to learning a weight vector along with the cut-off points between classes. For problems where this condition is not satisfied a more complex techniques using genetic programming techniques can be used.

References

[1] B.E. Flores, D.L. Olson and V.K. Dorai, Management of Multicriteria Inventory Classification, *Mathl. Comput. Modeling* Vol. 16, No. 12, pp. 71-82, 1992.

[2] B.E. Flores and D.C. Whybark, Multiple Criteria ABC Analysis, *International Journal of Operations and Production Management* Vol. 6, No. 3, pp. 38-46, 1986.

[3] D. E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Reading, Massachusetts, Addison-Wesley, 1989.

[4] T.L. Saaty, The Analytic Hierarchy Process, New York, McGraw-Hill, 1980.