A Learning Method for Developing PROAFTN Classifiers and a Comparative Study with Decision Trees

Nabil Belacel and Feras Al-Obeidat

Institute for Information Technology
National Research Council of Canada

Abstract. PROAFTN belongs to Multiple-Criteria Decision Aid (MCDA) paradigm and requires a several set of parameters for the purpose of classification. This study proposes a new inductive approach for obtaining these parameters from data. To evaluate the performance of developed learning approach, a comparative study between PROAFTN and a decision tree in terms of their learning methodology, classification accuracy, and interpretability is investigated in this paper. The major distinguished property of Decision tree is that its ability to generate classification models that can be easily explained. The PROAFTN method has also this capability, therefore avoiding a black box situation. Furthermore, according to the proposed learning approach in this study, the experimental results show that PROAFTN strongly competes with ID3 and C4.5 in terms of classification accuracy.

Keywords: Classification, PROAFTN, Decision Tree, MCDA, Knowledge Discovery.

1 Introduction

Decision tree learning is a widely used method in data mining and machine learning. The strength of decision trees (DT) can be summarized as: (1) Simple to understand and interpret. People are able to understand decision tree models after a brief explanation. (2) Not a black box model. The classification model can be easily explained by boolean logic. (3) The methodology used to construct a classification model is not hard to understand. (4) The classification results are usually reasonable. These advantages of DT make it a common and highly used classification method in research and applications [4].

This paper introduces a new learning technique for the classification method PROAFTN which requires several parameters (e.g intervals, discrimination thresholds and weights) that need to be determined to perform the classification. This study investigates a new automatic approach for the elicitation of PROAFTN parameters from data and prototypes during training process. The major characteristics of PROAFTN can be summarized as follows:

- PROAFTN is not a black box and the results are automatically explained, that is it provides the possibility of access to more detailed information concerning the classification decision.

C. Butz and P. Lingras (Eds.): Canadian AI 2011, LNAI 6657, pp. 56–61, 2011.

– PROAFTN can perform two learning paradigms: deductive and inductive learning. In the deductive approach, the decision maker has the role of establishing the required parameters for the studied problem, whereas in an inductive approach, the parameters and the classification models are obtained automatically from the datasets.

Based on what have been presented above, one can see that DT and PROAFTN can generate classification models which can be easily explained and interpreted. However, when evaluating any classification method there is another important factor to be considered: classification accuracy. Based on the experimental study presented in Section 4, PROAFTN can generate a higher classification accuracy than decision tree learning algorithms: ID3 and C4.5 [9].

The paper is organized as follows. Section 2 presents the PROAFTN method. Section 3 proposes automatic learning methods based on machine learning techniques to infer PROAFTN parameters and prototypes. In Section 4 a comparative study based on computational results generated by PROAFTN and DT (ID3 and C4.5) on some well-known datasets is presented and analyzed. Finally, conclusions and future works are presented in Section 5.

2 PROAFTN Method

PROAFTN procedure belongs to the class of supervised learning to solve classification problems. PROAFTN has been applied to the resolution of many real-world practical problems [6] [7] [10]. The following subsections describe the required parameters, the classification methodology, and the procedure used by PROAFTN.

2.1 Initialization

From a set of n objects known as a training set, consider a is an object which requires to be classified; assume this object a is described by a set of m attributes $\{g_1, g_2, ..., g_m\}$ and z classes $\{C^1, C^2, ..., C^z\}$. Given an object a described by the score of m attributes, for each class C^h, we determine a set of L_h prototypes. For each prototype b_i^h and each attribute g_j, an interval $[S_j^1(b_i^h), S_j^2(b_i^h)]$ is defined where $S_j^2(b_i^h) \geq S_j^1(b_i^h)$.

To apply PROAFTN, the intervals: the pessimistic $[S_j^1(b_i^h), S_j^2(b_i^h)]$ and the optimistic $[S_j^1(b_i^h) - d_j^1(b_i^h), S_j^2(b_i^h) + d_j^2(b_i^h)]$ should be determined prior to classification for each attribute. As mentioned above, the indirect technique approach will be adapted to infer these intervals. The following subsections explain the stages required to classify the object a to the class C^h using PROAFTN.

2.2 Computing the Fuzzy Indifference Relation

To use the classification method PROAFTN, we need first to calculate the fuzzy indifference relation $I(a, b_i^h)$. The calculation of the fuzzy indifference relation is based on the concordance and non-discordance principle which is identified by:

$$I(a, b_i^h) = \sum_{j=1}^{m} w_j^h C_j(a, b_i^h) \tag{1}$$

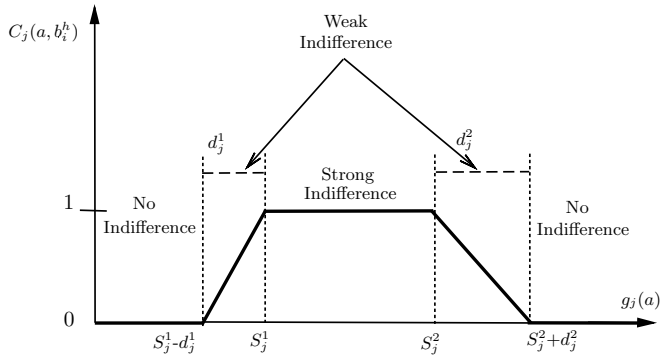

Fig. 1. Graphical representation of the partial indifference concordance index between the object a and the prototype b_i^h represented by intervals

where w_j^h is the weight that measures the importance of a relevant attribute g_j of a specific class C^h:

$$w_j \in [0,1] \text{ , and } \sum_{j=1}^{m} w_j^h = 1$$

$$j = 1,...,m; h = 1,...,z$$

$C_j(a,b_i^h)$ is the degree that measures the closeness of the object a to the prototype b_i^h according to the attribute g_j. To calculate $C_j(a,b_i^h)$, two positive thresholds $d_j^1(b_i^h)$ and $d_j^2(b_i^h)$ need to be obtained. The computation of $C_j(a,b_i^h)$ is graphically presented in Fig. 1.

2.3 Evaluation of the Membership Degree

The membership degree between the object a and the class C^h is calculated based on the indifference degree between a and its nearest neighbor in B^h. The following formula identifies the nearest neighbor:

$$d(a,C^h) = \max\{I(a,b_1^h), I(a,b_2^h),...,I(a,b_{L_h}^h)\} \tag{2}$$

2.4 Assignment of an Object to the Class

The last step is to assign the object a to the right class C^h; the calculation required to find the right class is straightforward:

$$a \in C^h \Leftrightarrow d(a,C^h) = \max\{d(a,C^i)/i \in \{1,...,z\}\} \tag{3}$$

3 Proposed Techniques to Learn PROAFTN

As discussed earlier, PROAFTN requires the elicitation of its parameters for the purpose of classification. Several approaches have been used to learn PROAFTN in [1] [2] [3].

Algorithm 1. Building the classification model for PROAFTN

Determine of a ***threshold*** β as reference for interval selection
$z \leftarrow$ Number of classes, $i \leftarrow$ Prototype's index
$m \leftarrow$ Number of attributes, $k \leftarrow$ Number of intervals for each attribute
$I^r_{jh} \leftarrow$ Apply discretization to get $\{S^{1r}_{jh}, S^{2r}_{jh}\}$ for each attribute g_j in each class C^h
$\Re \leftarrow$ Percentage of values within the interval I^r_{jh} per class
Generate PROAFTN intervals using discretization
for $h = \leftarrow 1, z$ **do**
    $i \leftarrow 0$
    for $g \leftarrow 1, m$ **do**
        for $r \leftarrow 1, k$ **do**
            if $\Re$ of $I^r_{jh} \geq \beta$ **then**
                Choose this interval to be part of the prototype b^h_i
                Go to next attribute g_{m+1}
            else
                Discard this interval and find another one (*i.e.,* I^{r+1}_{jh})
            end if
        end for
    end for
    if ($b^h_i \neq \emptyset \; \forall g_{jh}$) **then** $i \leftarrow i+1$
    end if
    (Prototypes' composition):
    The selected branches from attribute g_1 to attribute g_m represent the induced prototypes for the class C^h
end for

In this study however, a different technique is proposed to get these parameters from data. During the learning process, the necessary preferential information (a.k.a. prototypes) required to construct the classification model are extracted first; then this information are used for assigning the new cases (testing data) to the closest class. The PROAFTN parameters that are required to be elicited automatically from training dataset are: $\{S^1_j(b^h_i), S^2_j(b^h_i), d^1_j(b^h_i), d^2_j(b^h_i)\}$. This study proposes the discretization techniques to infer these parameters. Once these parameters are determined, the next stage is to build the classification model, which consists of a set of prototypes that represents each category. The obtained prototypes can then be used to classify the new instances.

Discretization techniques are utilized to obtain the intervals $[S^1_j(b^h_i), S^2_j(b^h_i)]$ automatically for each attribute in the training dataset. The obtained intervals will then be adjusted to get the other fuzzy intervals $[S^1_j(b^h_i) - d^1_j(b^h_i), S^2_j(b^h_i) + d^2_j(b^h_i)]$, which will be used subsequently for building the classification model.

Following to the discretization phase is model development stage. The proposed model uses an induction approach given in Algorithm 1. The tree is constructed in a top-down recursive manner, where each branch represents the generated intervals for each attribute. The prototypes can then be extracted for the decision tree to compose decision rules to be used for classifying testing data.

Table 1. Dataset Description

Dataset	Instances	Attributes	Classes
Breast Cancer	699	11	2
Heart Disease	303	14	2
Haberman's Survival	306	3	2
Iris	150	4	3
Mammographic Mass	961	4	2
Pima Diabetes	768	8	2
Vehicle	846	18	4
Vowel Context	990	11	10
Wine	178	13	3
Yeast	1484	8	10

4 Application of the Developed Algorithms

The proposed method was implemented in java applied to 10 popular datasets described in Table 1. These datasets are available on the public domain of the University of California at Irvine (UCI) [5]. To compare our proposed approaches with ID3 and C4.5 algorithms, we have used the open source platform Weka [11] for this purpose. The comparisons are made on all datasets using stratified 10-fold cross validation.

The generated results applied on the datasets for PROAFTN, ID3 and C4.5 (pruned and unpruned) is shown in Table 2. The Friedman test [8] is used to recognize the performance of PROAFTN against other DT classifiers.

Table 2. ID3 and C4.5 versus PROAFTN in terms of classification accuracy

	Algorithm / Dataset	ID3	C4.5 (unpruned)	C4.5 (pruned)	PROAFTN
1	Breast Cancer	89.80	94.56	94.56	97.18
2	Heart Disease	74.10	74.81	76.70	79.04
3	Haberman's Survival	59.80	70.92	71.90	70.84
4	Iris	90.00	96.00	96.00	96.57
5	Mammographic Mass	75.35	81.27	82.10	84.30
6	Pima Diabetes	58.33	71.22	71.48	72.19
7	Vehicle	60.77	72.93	72.58	76.36
8	Vowel context	72.42	82.63	82.53	81.86
9	Wine	80.5	91.55	91.55	97.33
10	Yeast	41.71	54.78	56.00	57.00
Avg		70.28	79.07	79.54	81.27
Rank		4	3	2	1

5 Discussion and Conclusions

The common advantages of the PROAFTN method and the DT could be summarized as: (i) Reasoning about the results, therefore avoiding black box situations, and (ii)

Simple to understand and to interpret. Furthermore, in this study PROAFTN was able to outperform ID3 and C4.5 in terms of classification accuracy.

To apply PROAFTN, some parameters should be determined before performing classification procedures. This study proposed the indirect technique by using discretization to establish these parameters from data.

It has been shown in this study that PROAFTN is a promising classification method to be applied in a decision-making paradigm and knowledge discovery process. Hence, we have a classification method that relatively outperforms DT and is also interpretable. More improvements could be made to enhance PROAFTN; this includes (i) involve the weights factor in the learning process. The weights in this paper are assumed to be equal; (ii) extend the comparative study to include various classification methods from different paradigms.

References

1. Al-Obeidat, F., Belacel, N., Carretero, J.A., Mahanti, P.: A Hybrid Metaheuristic Framework for Evolving the PROAFTN Classifier. Special Journal Issues of World Academy of Science, Engineering and Technology 64, 217–225 (2010)
2. Al-Obeidat, F., Belacel, N., Carretero, J.A., Mahanti, P.: Automatic Parameter Settings for the PROAFTN Classifier Using Hybrid Particle Swarm Optimization. In: Li, J. (ed.) AI 2010. LNCS, vol. 6464, pp. 184–195. Springer, Heidelberg (2010)
3. Al-Obeidat, F., Belacel, N., Carretero, J.A., Mahanti, P.: Differential Evolution for learning the classification method PROAFTN. Knowledge-Based Systems 23(5), 418–426 (2010)
4. Apteand, C., Weiss, S.: Data mining with decision trees and decision rules. Future Generation Computer Systems (13) (1997)
5. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007)
6. Belacel, N., Boulassel, M.: Multicriteria fuzzy assignment method: A useful tool to assist medical diagnosis. Artificial Intelligence in Medicine 21(1-3), 201–207 (2001)
7. Belacel, N., Vincke, P., Scheiff, M., Boulassel, M.: Acute leukemia diagnosis aid using multicriteria fuzzy assignment methodology. Computer Methods and Programs in Biomedicine 64(2), 145–151 (2001)
8. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30 (2006)
9. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)
10. Sobrado, F.J., Pikatza, J.M., Larburu, I.U., Garcia, J.J., de Ipiña, D.: Towards a clinical practice guideline implementation for asthma treatment. In: Conejo, R., Urretavizcaya, M., Pérez-de-la-Cruz, J.-L. (eds.) CAEPIA/TTIA 2003. LNCS (LNAI), vol. 3040, pp. 587–596. Springer, Heidelberg (2004)
11. Witten, H.: Data Mining: Practical Machine Learning Tools and Techniques. Kaufmann Series in Data Management Systems (2005)