Decision Tree Classification Implementation with Fuzzy Logic

Renuka D. Suryawanshi *, D. M. Thakore**

****Computer Engineering Department Year Bharti Vidyapeeth, Pune,India***
***** Computer Engineering Department Bharti Vidyapeeths Deemed University College Of Engineering,Pune,India***

Summary:
Data mining is having an aim to analyze the observation datasets to find relationship and to present the data in ways that are both understandable and usable. This paper basically focuses on the classification technique of datamining to identify the class of an attribute with an ID3 (classical decision tree approach) and then to add fuzzification to improve the result of ID3.It also contains design and implementation of this combined approach with chosen datasets. Id3 results are based on information gain theory and Entropy values of each attribute. Fuzzy ID3 results are based on information gain of fuzzy dataset and fuzzy entropy.
Classification results are presented as decision tree which incorporates the result of Id3 & FID3.
Keywords:
ID3(Iterative Dichotomizer 3),FID3(Fuzzy iterative dichotomizer3),CLS(ConceptLearningSystem),IG(Information Gain)

1. Introduction

1.1 Decision Tree

In data mining and machine learning, decision tree is a predictive model that is mapping from observations about an item to conclusions about its target value. The machine learning technique for inducing a decision tree from data is called decision tree learning.[2]

'Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Decision tree learning is one of the most widely used and practical methods for inductive inference'.

Decision tree learning algorithm has been successfully used in expert systems in capturing knowledge. The main task performed in these systems is using inductive methods to the given values of attributes of an unknown object to determine appropriate classification according to decision tree rules.[1]

2. ID3(Iterative Dichotomizer 3)

Iteractive Dichotomizer 3 algorithm [1] is one of the most used algorithms in machine learning and data mining due to its easiness to use and effectiveness. J. Rose Quinlan developed it in 1986 based on the Concept Learning System (CLS) algorithm. It builds a decision tree from some fixed or historic symbolic data in order to learn to classify them and predict the classification of new data. The data must have several attributes with different values. Meanwhile, this data also has to belong to diverse predefined, discrete classes (i.e. Yes/No). Decision tree chooses the attributes for decision making by using information gain (IG). [3]

2.1 Implementation of ID3

Here we implemented the ID3 in java as per following steps.

Step 1: we have chosen the following sample dataset as an example to show the result.

Table 1: Dataset

Column 0	Column 1	Column 2	Column 3	Column 4
Male	0	Cheap	Low	Bus
Male	1	Cheap	Medium	Bus
Female	1	Cheap	Medium	Train
Female	0	Cheap	Low	Bus
Male	1	Cheap	Medium	Bus
Male	0	Standard	Medium	Train
Female	1	Standard	Medium	Train
Female	1	Expensive	High	Car
Male	2	Expensive	Medium	Car
Female	2	Expensive	High	Car

Step 2: Information gain and Entropy calculation

The basic ID3 method selects each instance attribute classification by using statistical method beginning in the top of the tree. The core question of the method ID3 is how to select the attribute of each pitch point of the tree. A statistical property called information gain is defined to measure the worth of the attribute. The statistical quantity Entropy is applied to define the information gain, to choose the best attribute from the candidate attributes.

The definition of Entropy is as follows:

$$\mathbf{H(S)=\sum_{I}^{N} - P_i * log_2(P_i)} \quad \textbf{(1)}$$

where Pi is the ratio of class C_i in the set of examples $S = \{x_1,x_2,...x_K\}$

Manuscript received October 5, 2012
Manuscript revised October 20, 2012

Step 3: Calculating highest information gain. Now consider the above dataset shown in Table 2.1.As this algorithm is based upon information gain of each attribute with entropies. First we need to calculate the information gain for each record set according to attributes given in dataset.

(i) Calculating gain for column 0 record set:
[Male, Male, Female, Female, Male, Male, Female, Female, Male, Female]. For
Male: Record set: [Bus, Bus, Bus, Train, Car]
Entropy: 0.44217935649972373
For **Female**: Record set: [Train, Bus, Train, Car, Car]
Entropy:0.46438561897747244

(ii) Getting impurity using entropy
Record set: [Bus, Bus, Train, Bus, Bus, Train, Train, Car, Car, Car]
Entropy: 0.5287712379549449
gain:0.22108967824986187
gain:0.4532824877385981
gain:0.4532824877385981
Information gain: 0.07548875021634677

As per the same way we calculated the information gain for each record set.
Below we give the calculation for each record set
Column 1 :
Calculating gain for record set: [0, 1, 1, 0, 1, 0, 1, 1, 2, 2]
Information gain: 0.338082178394093

Column 2 :
Calculating gain for record set: [Cheap, Cheap, Cheap, Cheap, Cheap, Standard, Standard, Expensive, Expensive, Expensive]
Information gain: 0.6321928094887361

Column 3 :
Calculating gain for record set:[Low, Medium, Medium, Low, Medium, Medium, Medium, High, Medium, High]
Information gain: 0.253282487738598

Step 4 :Generate Information Gain Table
[0.07548875021634677, 0.338082178394093, 0.6321928094887361, 0.253282487738598]
Highest gain column: 0.6321928094887361
As per the result we get highest information gain for **column no. 2**
We design a decision tree based upon the results as we get the highest information gain on column no. 2.
The decision tree is given below.

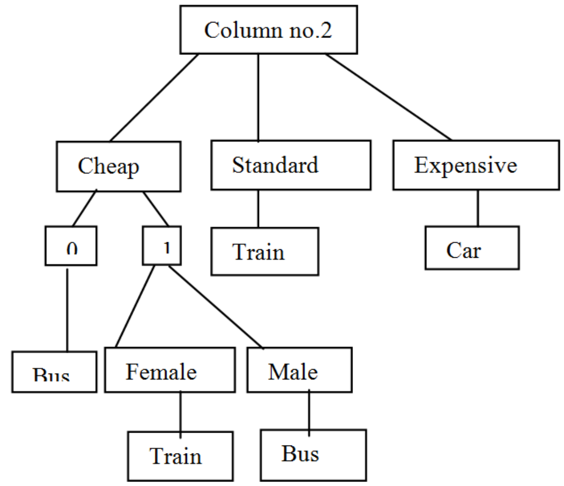

Fig. 1 Decision tree based on ID3

3. Fuzzy logic

The concept of Fuzzy Logic (FL) was given by Lotfi Zadeh, a professor at the University of California at Berkley, and presented as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership (Babuska, 1998). It is basically a multi-valued logic that allows intermediate values to be defined between conventional evaluations.

3.1 Fuzzy sets

A fuzzy set is a set without a crisp and clearly-defined boundary or without binary membership characteristics. Unlike an ordinary set where each object (or element) either belongs or does not belong to the set, fuzzy set can contain elements with only a partial degree of membership. In other words, there is a 'softness' associated with the membership of elements in a fuzzy set.

An example of a fuzzy set could be 'the set of tall people.' There are people who clearly belong to the above set and others that cannot be considered as tall. Since the concept of 'tall' is not precisely defined (for example, >2m), there will be a gray zone in the associated set where the membership is not quite obvious. As another example, consider the variable 'temperature'. It can take a fuzzy value (e.g., cold, cool, tepid, warm, and hot).Each fuzzy value such as 'hot' is called a fuzzy descriptor. It may be represented by a fuzzy set because any temperature that is considered to represent 'hot' belongs to this set and any other temperature does not belong to the set. A crisp set or a precise temperature interval such as 25 °C to 30 °C cannot be indentified to represent warm temperatures.

3.2 Membership function

A fuzzy set can be represented by a membership function. This function indicates the grade (degree) of membership within the set, of any element of the universe of discourse (e.g. the set of entities over which certain variables of interest in some formal
treatment may range).

The membership function maps every element of the universe to numerical values in the interval [0, 1]. Specifically,

$$\boldsymbol{\mu_A(x) : X \rightarrow [0, 1]} \qquad \textbf{(2)}$$

Where $\mu_A\ (x)$ is the membership function of the fuzzy set *A* in the universe *X*. Stated in another way, fuzzy set *A* is defined as a set of ordered pairs:

$$\boldsymbol{A = \{\ (x, \mu_A(x))\ ;\ x \in X, \mu_A\ (x) \in [0, 1]\}} \qquad \textbf{(3)}$$

The membership function (x) represents the grade of possibility that an element *x* belongs to the set *A*. It is a curve that defines how each point in the input space is mapped to a membership value. A membership function value of zero indicates that corresponding element is definitely not an element of the fuzzy set. A membership function value of unity implies that the corresponding element is definitely an element of the fuzzy set. A grade of membership greater than 0 and less than 1 corresponds to a non-crisp (or fuzzy) membership, and the corresponding elements fall on the fuzzy boundary of the set. The closer the $\mu A\ \ x$ is to 1 the more the x is considered to belong to *A*, and similarly, the closer it is to 0 the less it is considered to belong to *A*.

The following is a summary of the characteristics of fuzzy set and membership function :

Fuzzy sets describe vague concepts (e.g., fast runner, hot weather, and weekend days).

A fuzzy set admits the possibility of partial membership in it. (e.g., Friday is sort of a weekend day, the weather is rather hot).

The degree to which an object belongs to a fuzzy set is denoted by a membership value between 0 and 1. (e.g., Friday is a weekend day to the degree 0.8).

A membership function associated with a given fuzzy set maps an input value to its appropriate membership value.

4. Fuzzification to create the decision tree

Fuzzy decision tree is an extension of classical decision tree and an effective method to extract knowledge in uncertain classification problems. It applies the fuzzy set theory to represent the data set and combines tree growing and pruning to determine the structure of the tree.

In general, there exist two different kinds of attributes: discrete and continuous. Many algorithms require data with discrete value. It is not easy to replace a continuous domain with a discrete one. This requires some partition and clustering. It is also very difficult to define the boundary of the continuous attributes. For example, how do we define whether the traffic-jam is long or short? Can we say that the traffic-jam of 3 km is long, and 2.9 km is short? Can we say it is cool when the temperature is 9, and it is mild for 10?

Therefore, some scholars quote the fuzzy concept in the method ID3, substitute the sample data with the fuzzy expression and form the fuzzy ID3 method.

Below is the example of the fuzzy representation for the sample data.

4.1 Fuzzy Entropy and Information Gain

Next, we have to calculate the fuzzy Entropy and Information Gain of the fuzzy data set to expand the tree.

In this case, we get the same result of the entropy of the as ID3

The formulas of the entropy for the attributes and the Information Gain are a little bit different because of the data fuzzy expression. Their definitions are
defined as follow respectively with the assumption dataset *S = {x_1, x_2,…. x_j)*

$$\mathbf{H_f(S,A)=} \quad \sum_{i=1}^{C} \frac{\sum_j^N \mu_{ij}}{S} \quad \mathbf{\log_2} \quad \frac{\sum_j^N \mu_{ij}}{S} \qquad \textbf{(4)}$$

$$\mathbf{G_f(S,A)=H_f(S)-} \quad \sum_{v\in A}^{N} \frac{|S_v|}{S} \quad \mathbf{{*}H_{fs}(sv,A)} \qquad \textbf{(5)}$$

where:

μ_{ij} is the membership value of the j^{th} pattern to the i^{th} class.

$H_f(S)$ presents the entropy of the set *S* of training examples in the node.

$|S_v|$ is the size of the subset *S*, S_v is the subset of of training examples x_j with v attribute.

$|S_v|$ presents the size of set S.

4.1.1Calculating highest gain

Following are the calculations for fuzzy gain and entropy

(i) Column [0] :

Calculating gain for record set:[Male, Male, Female, Female, Male, Male, Female, Female, Male,Female]

ForMale>>

Record set: [Bus, Bus, Bus, Train, Car]>>**Entropy:** 1.3709505944546687

ForFemale>>
Record set: [Train, Bus, Train, Car, Car]>>**Entropy**:1.5219280948873621
Getting impurity using entropy
Record set: [Bus, Bus, Train, Bus, Bus, Train, Train, Car, Car, Car]>>**Entropy:**1.5709505944546684
gain:0.6854752972273344
gain:1.4464393446710155
gain>>>1.4464393446710155
Information gain:0.12451124978365291
Same way we need to calculate the entropy and information gain for next record sets then we get
(ii) Column [1]
Calculating gain for record set:[0, 1, 1, 0, 1, 0, 1, 1, 2, 2]
Information gain: 0.5344977967946405

(iii) Column [2]
Calculating gain for record set:[Cheap, Cheap, Cheap, Cheap, Cheap, Standard, Standard, Expensive, Expensive, Expensive]
Information gain: 1.2099865470109874

Column [3]
Calculating gain for record set:[Low, Medium, Medium, Low, Medium, Medium, Medium, High, Medium, High]
Information gain: 0.6954618442383218

4.1.2 Information Gain Table

Next we need to create the information table of FID3
[0.12451124978365291, 0.5344977967946405, 1.2099865470109874, 0.6954618442383218]
From this table we get the following information
Highest gain column: 1.2099865470109874: column no. 2 Now we can use it to expand the tree.

4.1.3 Define thresholds

If the learning of FDT stops until all the sample data in each leaf node belongs to one class, it is poor in accuracy. In order to improve the accuracy, the learning must be stopped early or termed pruning in general. As a result, two thresholds are defined [8].

(i) Fuzziness control threshold θ_r

If the proportion of a data set of a class C_k is greater than or equal to a threshold θ_r, stop expanding the tree.
For example: if in sub-dataset the ratio of class 1 is 90%, class 2 is 10% and θ_r is 85%, then stop expanding.

(ii) Leaf decision threshold θ_n

If the number of a data set is less than a threshold θ_n , stop expanding.

For example, a data set has 600 examples where θ_n is 2%. If the number of samples in a node is less than 12 (2% of 600), then stop expanding.
The level of these thresholds has great influences on the result of the tree. We define them in different levels in our experiment to find optimal values.
Moreover, if there are no more attributes for classification, the algorithm does not create a new node.

4.1.4 The procedure to build the fuzzy decision tree: [8]

Create a *Root* node that has a fuzzy set of all data with membership value 1. With the result of the calculation above, we use the attribute *column no.2* to expand the tree.

Generate two sub-nodes with the examples, where the membership values at these sub-nodes are the product of the original membership values at *Root* and the membership values of the attribute *column no.2*. The example is omitted if its membership value is null.

In this way we have implemented the basic decision tree and fuzzy ID3 algorithm in java.
And the results of implementation is presented in terms of information gain ,Entropy and decision tree.

5. Comparing the algorithms among ID3, FID3 and PFID3

Because FID3 and PFID3 are based on ID3, these three methodologies have similar algorithms. However, there also exist some differences.
(i) Data representation: The data representation of ID3 is crisp while for FID3 and PFID3, they are fuzzy, with continuous attributes. Moreover, the membership functions of PFID3must satisfy the condition of *well-defined sample space* . The sum of all the membership values for all data value *i x* must be equal to 1.
(ii)Termination criteria: ID3: if all the samples in a node belong to one class or in other words, if the entropy equals to null, the tree is terminated. Sometimes, people stop learning when the proportion of a class at the node is greater than or equal to a predefined threshold. This is called pruning. The pruned ID3 tree tops early because the redundant branches have been pruned. FID3 & PID3: there are three criteria's.
1) If the proportion of the dataset of a class is greater than or equal to a threshold θr
2) If the number of a data set is less than another threshold θn
3) If there are no more attributes at the node to be classified If one of these three criteria's is fulfilled, the

learning is terminated.

6. Conclusion

We have given the results of ID3, FID3 from this we conclude that applying the *well-defined sample space* to the fuzzy partition have a positive effect on the performance. Fuzzy decision tree is an extension of classical decision tree to extract the knowledge in uncertain classification problems The main difference between the learning ID3 and FID3 is that for discrete values ID3 works well but for continues values it will not give more accuracy ,Fuzzy ID3 resolves this problem of uncertainty as per the values shown..

Classical decision tree has two issues like How to choose the best criteria to split the training instances and what the stopping rule is to terminate the splitting procedure.

These two issues are solved with the help of fuzzy decision tree.

On the contrary, the data point of FID3 can be overweight or underweight. Thus, the learning is inaccurate due to the imbalanced weight of the data.This will be the future scope for further enhancement.

We implemented the system which is based on ID3 and FID3 and result are shown in experiments and results part.

References:

[1] An Implementa tion of ID3 Decision Tree Learning Algorithm Wei Peng, Juhua Chen and Haiping Zhou Project of Comp 9417: Machine Learning University of New South Wales, Schoo l o f Computer Science & Engineering,Sydney, NSW 2032, Australia weipengtig er@hotmail.com

[2] http://www.roselladb .com

[3] J. Ha n and M. Kamber. Data Mining: Concepts and Techniq ues. Morgan Kaufmann, 2000

[4] Top 10 algorithms in data mining Received: 9 July 2007 / Revised: 28 September 2007 / Accepted: 8 October 2007Published online: 4 December 2007 © Springer-Verlag London Limited 2007

[5] Ahmed S, Coenen F, Leng PH (2006) Tree-based partitioning of date for association rule mining. KnowlInf Syst 10(3):315 –33 1

[6] Bloch DA, Olshen RA, Walker MG (2002) Riskestima tion fo r classificatio n trees. J Comput Graph Sta t11:263–288

[7] Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. adsworth,Belmont

[8] Breiman L(1968) Probability theory. Addison-Wesley, Reading. Republished(1991) in Classics of mathematics.SIAM, Philad elphia.

[9] [9] O. Cord´o n, F. Gomide, F. Herrera, F. Hoffmann, a nd L. Magdalena. Ten years of genetic fuzzy systems:Current framework and new trends. Fuzzy Sets and Systems141: 5–31. 2001.

[10] W. W. Cohen. Fast effective rule inductio n. In Machine Lea rning: Proceedings ofthe 12th Interna tional Conference, Morga n Kaufmann, pp. 115–1 23. 199 5.

[11] R. B. Bhatt and M. Gopal. FRID: Fuzzyrough interactive dichotomizers. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'0 4). Piscataway, NJ: IEEE Press, p p. 1 337–1342. 2004

[12] Knowledge Discovery using Text Mining: A Programmab le Implementation on Information Extraction and Categorization by Atika Mustafa, Ali Akbar, and Ahmer Sulta n Na tiona l University of Computer and Emerging Sciences-FAST, International Journal of Multimedia and Ub iquito us Engineering Vol. 4, No. 2

[13] Differentiating Data and Text-Mining Termino logy JAN H. KROEZE, MACHDEL C. MATTHEE AND THEO J.D. BOTHMAUniversity of Pretoria.

[14] Automated Learning of Decisio n Rules for Text Categorizatio n C. Apte, F. n Damerau, and S.M. Weiss ACM Transactions on Info rmation Systems, 1994

[15] Text Classification in Information Retrieval using Winnow P.P.T.M. van Mun Department of Computing Science, Catholic University of Nijmegen Toernooiveld 1, NL-6 525 ED, Nijmegen, The Netherla nds

[16] Von Altrock, Constantin (1995). Fuzzy logic and Neuro Fuzzy applications expla ined. Upper Sad dle River, NJ: Prentice Hall PTR. ISBN 0-13-368465-2 . Zimmerma nn, H. (2001). Fuzzy set theory and its applications. Boston: Kluwer Acad emic Pub lishers. ISBN 0-7923-743 5-5 .

[17] "Building Decision Trees with the ID3 lgorithm", by: Andrew Colin, Dr. Dobbs Journal, June 19 96

[18] HAN, J. AND KAMBER, M. 2001. Data mining: concepts and techniques. Morgan Kaufmann, San Francisco, CA.

[19] Von Altrock, Constantin(1995). Fuzzy logic and NeuroFuzzy applications exp lained. Upper Saddle River,NJ: Prentice Hall PTR. ISBN 0-13-368465-2.

[20] Zimmermann, H. (2001). Fuzzy set theory and its applications. Boston: Kluwer Academic Publishers. ISBN 0-7923-7435-5.

[21] Implementation of Anomaly Detection Technique Using Machine Learning Algorithms ISSN 2047-3338 International Journal of Computer Science and Telecommunications [Volume 2, Issue 3, June 2011]

[22] The ID3 Algorithm. Department of Computer and Information Science and Engineering of University of Florida.http://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm on 15. Nov. 2005.