NTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY & CREATIVE ENGINEERING (ISSN:2045-8711)
VOL.1 NO.6 JUNE 2011

Coefficient of Variation based Decision Tree (CvDT)

Hima Bindu K [#1], Swarupa Rani K [#2], Raghavendra Rao C [#3]

[#] *Department of Computer and Information Sciences, University of Hyderabad*
Hyderabad, 500046, India

[1] himagopal@gmail.com,[2] swarupacs@uohyd.ernet.in,[3] crrcs@uohyd.ernet.in

***Abstract*— Decision trees are widely used for classification. Several approaches exist to induce decision trees. All these methods vary in attribute selection measures i.e., in identifying an attribute to split at a node. This paper proposes a novel splitting criteria based on Coefficient of Variation and it is named as Coefficient of Variation Gain (CvGain). The decision trees built with CvGain are compared with those built with Entropy and Gainfix. Empirical analysis based on standard data sets revealed that Coefficient of Variation based decision tree (CvDT) has less computational cost and time.**

Keywords: Coefficient of Variation; CvGain; Decision Tree; Splitting Criteria

I. INTRODUCTION

Decision trees are well known for classification [6]. Decision trees are easy to interpret and they simplify the complex decision making process. ID3 [10], C4.5 [11], CART [2] are few popular implementations of decision trees. ID3 algorithm is the first decision tree implementation. Building a decision tree follows a greedy approach for choosing the best attribute for splitting at a node. Splitting criteria plays a vital role in building a decision tree. Information Gain, Gain ratio, Gini index, Chi square statistics and Kappa index are the well known splitting criteria. Coefficient of Variation (Cv) [13], is a measure of consistency of a distribution and is used in applied domain. The application of Cv for constructing risk trees in managerial studies is demonstrated in [3]. Cv has not attracted the researchers of data mining as a splitting criteria till date. Cv is a normalized measure of dispersion of a probability distribution. This paper proposes building a decision tree using Cv. ID3, proposed by Quinlan uses Information gain for attribute selection, which is based on information theory. But Information Gain is biased towards multi-valued attributes. C4.5 is a successor of ID3 and uses gain ratio, which is an extension of Information gain. Gain ratio overcomes the biasing for multi-valued attributes by applying some normalization to Information Gain. But Gain ratio tends to prefer unbalanced splits. The Gini index which considers a binary split for each attribute is used by CART. But Gini index also prefers multi-valued attributes and has a difficulty in dealing large number of classes. The limitations of impurity based measures like Information gain and Gini index are given by [5]. They have proposed a class of attribute selection measures called C-SEP to overcome those limitations.

Reference [7] has applied ID3 on the reduced data obtained by reduct attributes based on rough set theory [9]. Reduct selects only predominant attributes. Thus one can achieve dimension reduction. It is reported that the resulting tree generates fewer classification rules with comparable classification accuracy to ID3. Reference [12] has built a reduct based decision tree where the splitting attributes are selected according to their order of presence in the reduct. To the best of our knowledge, the latest splitting criteria used was based on Kappa index as proposed in [4]. They proposed fixed information gain, called as Gainfix , as the new standard for selecting splitting attributes. Gainfix considers relationship between condition attributes and decision attributes in addition to Information Gain. They claimed that the decision tree (which is named as FID3 by them) built using Kappa achieves better performance and simpler decision tree than ID3.

Since its inception, ID3 has been thoroughly studied by various researchers. ID3 uses Information Gain as the splitting criteria. But Information Gain uses frequencies ignoring its ordinates and is based on the Entropy which invokes logarithmic function several times. The computation of Cv is less expensive as it uses simple arithmetic operations and square root function. This contrasting feature inspired the present study to use Cv for construction of decision trees. The tree built based on the Cv will be called as Cv based Decision Tree, in short CvDT. As the computational complexity of Cv is low, it is expected that Cv based decision tree construction will take less time. This is proved by the hypothesis test performed with paired t-

test. Hence it is suitable for agent based applications, where a decision tree has to be built in real time.

The performance characteristics have been tabulated using tenfold cross validation test on the proposed CvDT as well as on ID3 and FID3 methods for evaluation purpose.

This paper is organized as follows: Section 2 illustrates Coefficient of variation and introduces CvGain along with computations. The CvDT algorithm is discussed in Section 3. Section 4 illustrates CvDT construction with a simple example. Section 5 describes the data sets considered for validation and the adopted validation procedure. Section 6 brings out contrasting features of CvDT and ID3. The paper concludes with section 7.

II. CVGAIN

A. Coefficient of Variation

Coefficient of Variation [13], [1] is the ratio of standard deviation σ and mean μ.

$$Cv = \frac{\sigma}{\mu} \qquad (1)$$

Coefficient of Variation is a dimensionless number and hence it is suitable for comparing data in different units or with widely different means. Cv is defined for non zero mean. The computation of Cv is illustrated with Table I data. This data contains two attributes: High School GPA (called as A1) and College GPA (called as A2).

TABLE I
GPA DATA

Student	A1	A2	D
S1	3	2	2
S2	3	1	3
S3	4	3	1
S4	2	1	3
S5	3	3	2

The computations of Cv for each attribute of GPA data are given below.

Cv(A1)=σ(A1)/μ(A1)*100 = (0.6325/3)*100 = 21.0819
Cv(A2)=σ(A2)/μ(A2) *100 = (0.8944/2)*100 = 44.7214
Cv(D)=σ(D)/μ(D) *100 = (0.7483/2.2)*100 = 34.0151

B. CvGain

Let DT be the decision table which is preprocessed such that Cv can be computed.

Let DT = [A1, A2, A3… An, D] where A1, A2….An are the conditional attributes and D is the decision attribute.

Coefficient of Variation of decision attribute D is given by

$$Cv(D) = \frac{\sigma(D)}{\mu(D)} * 100 \qquad (2)$$

Coefficient of Variation of D conditioned on Ai having 'v' distinct values (a1, a2 … av) is given by

$$Cv(D|Ai) = \sum_{j=1}^{v} P_j Cv(D \mid A_i = a_j) \qquad (3)$$

Where aj is the jth possible value of Ai with chance Pj And

$$CvGain(Ai) = Cv(D) - Cv(D|Ai) \qquad (4)$$

Using GPA data (Table II), the detailed computations of CvGain are given below along with the conditional tables for Cv (D|A).

TABLE II
CONDITIONAL TABLE WITH A1 = 2

Student	A1	D
S4	2	3

From Table II, Cv(D|A1 = 2) = 0/3 * 100 = 0.

TABLE III
CONDITIONAL TABLE WITH A1 = 3

Student	A1	D
S1	3	2
S2	3	3
S5	3	2

From Table III, Cv(D|A1=3)= 0.4714/ 2.33*100 = 20.20

TABLE IV
CONDITIONAL TABLE WITH A1 = 4

Student	A1	D
S3	4	1

From Table IV, Cv(D|A1 = 4) = 0/1 * 100 = 0
Assuming that P (Ai = aj) is the probability that attribute Ai takes the value aj,
Cv(D|A1)=P(A1=2)*Cv(D|A1=2)+P(A1=3)*Cv(D|A1=3) + P(A1=4)*Cv(D|A1=4)
Hence,
Cv(D|A1)=1/5*0 + 3/5*20.2031 + 1/5*0 = 12.1219
CvGain(A1) = 34.0151 - 12.1219 = 21.8932
With similar calculations for attribute A2,
Cv(D | A2 = 1) = 0 /3 *100 = 0
Cv(D | A2 = 2) = 0/2 *100 = 0
Cv(D | A2 = 3) = 0.5/1.5 *100 = 33.3333
Cv(D|A2) = 2/5*0 + 1/5*0 + 2/5*33.33 = 13.33
CvGain(A2) = 34.0151 - 13.3333 = 20.6818

As CvGain(A1) is large when compared with CvGain(A2), A1 is selected as the splitting attribute.

C. Algorithm

Algorithm CvDT: Generate CvDT from the decision table DT.
Input: Decision Table DT with *attribute_list* and decision attribute D.
Output: CvDT
Method:
(1) create a node N;
(2) **if** Cv(D)=0 **then**
(3) **return** N as a leaf node labeled with the class C, the class of all tuples;
(4) **if** *attribute_list* is empty **then**
(5) **return** N as a leaf node labeled with the *majority_class* in D; *//majority voting*
(6) *splitting attribute* = max(**CvGain** (*attribute_list*))
(7) *attribute_list* = *attribute_list* – *splitting_ attribute*;
(8) **for** each value j of *splitting_ attribute* *//partition the //tuples and grow sub trees for each partition*
(9) DT_j = { tuples in DT with *splitting_ attribute* = j };
(10) **if** $DT_j = \phi$ **then**
(12) create a leaf node labeled with majority class in DT_j and attach it to node N;
(13) **else** attach the node returned by **CvDT**(DT_j, *attribute_ list*) to node N;
(14) **end for**
(15) **return** N;

III. ILLUSTRATION

The popular Weather data set for the concept *Play Tennis* [8)] is considered for illustration purpose(Table V).

TABLE V
DECISION TABLE FOR THE CONCEPT "PLAY TENNIS"

Outlook	Temperature	humidity	Windy	Class
Overcast	Hot	high	False	N
Overcast	Mild	high	True	N
Overcast	Hot	normal	False	N
Overcast	Cool	normal	True	N
Rain	Mild	high	False	N
Rain	Mild	high	True	P
Rain	Cool	normal	False	N
Rain	Mild	normal	False	N
Rain	Cool	normal	True	P
Sunny	Hot	high	False	P
Sunny	Mild	high	False	P
Sunny	Hot	high	True	P
Sunny	Cool	normal	False	N
Sunny	Mild	normal	True	N

To compute Cv, the mean value need to be non zero. Hence the data need to be pre-processed in such a way that avoids 'mean' to be zero. A simple pre processing which assigns positive integers is used here for illustration. In fact any pre processing technique which gives non zero mean is applicable. Table VI shows the pre-processed data and Table VII shows the CvGain values.

TABLE VI
PREPROCESSED DECISION TABLE

Outlook	Temperature	humidity	windy	Class
2	3	2	1	2
2	2	2	2	2
2	3	1	1	2
2	1	1	2	2
1	2	2	1	2
1	2	2	2	1
1	1	1	1	2
1	2	1	1	2
1	1	1	2	1
3	3	2	1	1
3	2	2	1	1
3	3	2	2	1
3	1	1	1	2
3	2	1	2	2

TABLE VII
CVGAIN VALUES

Attribute	CvGain
Outlook	5.73
Temperature	0.45
Humidity	2.42
Windy	0.74

From the Table VII, Outlook has maximum CvGain; hence the attribute Outlook is selected as splitting attribute at root node. Hence the data will be split into three sub tables based on the Outlook values. For Outlook = 1, 2 and 3 the decision tables are shown in VIII, IX and X respectively.

TABLE VIII
DECISION TABLE FOR OUTLOOK = 1

Temperature	humidity	Windy	class

2	2	1	2
2	2	2	1
1	1	1	2
2	1	1	2
1	1	2	1

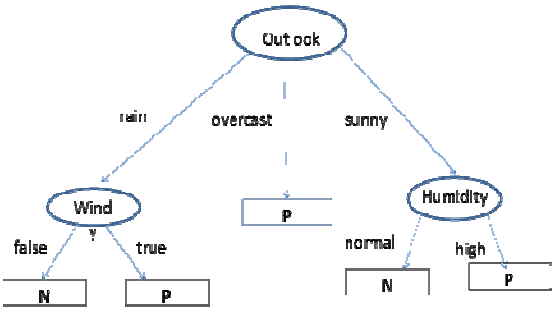

Fig. 1. Final Decision tree.

TABLE IX
DECISION TABLE FOR OUTLOOK = 2

Temperature	Humidity	windy	Class
3	2	1	2
2	2	2	2
3	1	1	2
1	1	2	2

TABLE X
DECISION TABLE FOR OUTLOOK = 3

Temperature	humidity	windy	class
3	2	1	1
2	2	1	1
3	2	2	1
1	1	1	2
2	1	2	2

The corresponding building component of the decision tree is as shown in figure 1:

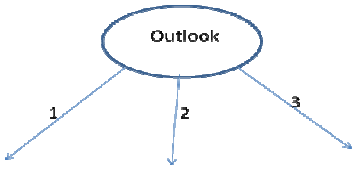

Fig. 1. Decision tree with Outlook as splitting criteria at root node.

With similar computations on tables VIII,IX and X, the decision tree is obtained as shown in figure 2 with preprocessed codes replaced with their original values.

IV. EXPERIMENT

To examine the effectiveness of our splitting criteria on decision tree construction, we collected ten datasets from UCI machine learning repository, shown in Table XI.

TABLE XI
CHARACTERISTICS OF DATA SETS

S.No	Data set	Number of Objects	Number of Attributes
1	Iris	150	4
2	Wine	178	13
3	Breast cancer	699	10
4	Blood Transfusion	748	4
5	Abalone	4177	8
6	Ecoli	336	7
7	Yeast	1484	8
8	Page-blocks	5473	10
9	Wine red	1599	11
10	Pima-Indians	768	8

We built decision trees with three different splitting criteria: Information Gain of ID3, Gainfix of FID3 and CvGain proposed in this paper. The data sets with continuous values are discritized. When the data set is nominal integer codes are used. The data sets are randomly permuted and tenfold cross validation is administered. Each time the same partitions of the data sets are used for building and testing the decision trees. The philosophy of constructing decision tree algorithm is the same for all the three trees, only with difference in the selection criteria. Information Gain, GainFix and CvGain are used as the attribute selection criteria for ID3, FID3 and CvDT respectively. We computed the classification performance and the times taken for training the decision trees as well as for testing them. We performed t-test to verify the statistical significance of our results (we used a standard significance level of 0.05). The characteristics of the datasets are shown in the table XI. The datasets collected contain 150 tuples

as the least and 5473 tuples as the highest. The least number of attributes taken is 4 while the highest is 13. The results are shown in table XII.

The advantage of CvGain is revealed in the times taken for decision tree generation. The generation times of CvDT are statistically significantly low when compared to the other two methods. In the case of the larger datasets considered in this experiment like Abalone and Page-blocks, the reduction of time is more clearly visible. With Abalone, 338 and 1672 milliseconds of time is saved when compared with ID3 and FID3 respectively. Similarly with Page-blocks they are 118 and 2726 milliseconds. Hence it is expected that CvGain

TABLE XII

COMPARISON OF ACCURACY, TIMES FOR CVDT, ID3 AND FID3

Data	Classification Performance			Generation Time in ms			Testing Time in ms		
	CvDT	ID3	FID3	CvDT	ID3	FID3	CvDT	ID3	FID3
Iris	97.33	97.33	97.33	20.25	23.30	43.85	0.06	0.05	0.05
Wine	95.56	97.22	96.67	50.71	76.91	226.08	0.07	0.06	0.06
Breast cancer	99.43	99.86	99.71	68.39	92.77	280.32	0.24	0.22	0.23
Blood Transfusion	81.81	81.67	81.81	131.06	153.24	264.64	0.34	0.33	0.33
Abalone	85.80	85.76	85.76	1586.44	1924.62	3258.47	18.01	18.26	21.10
Ecoli	95.88	95.00	95.29	123.71	168.67	319.79	0.13	0.13	0.13
Yeast	92.16	92.30	92.50	792.69	1002.69	1975.08	5.41	5.31	5.27
Page-blocks	97.15	97.28	97.44	785.26	903.79	3511.66	18.95	18.80	18.30
Wine red	95.63	95.19	95.31	756.97	1071.02	2741.15	5.39	5.51	5.07
Pima-Indians	95.97	96.23	96.36	265.04	355.26	840.57	2.34	2.33	2.33

is suitable for applications which require the decision trees to be built in real time.

The Classification Performances are more or less equal for all the three trees. The observations based on the experiment are as follows:

1. Basically all the three trees are working on the same partitions of the data in each the ten folds used in the experiment.
2. The same procedure is used to build the decision tree, with the variation in the splitting criteria.
3. The gain values of the attributes are different values with CvGain, Information Gain and GainFix. But the attribute with maximum gain value is the same with all the three methods for some of data sets. It is different for some of the data sets also.
4. It is possible that more than one attribute can have the maximum Gain value, and one of them is selected arbitrarily.
5. When the decision trees are verified, the decision trees built are the same for few data sets (Tom Mitchell, Iris, Blood Transfusion,) but different with all the other data sets. For some of the data sets even though the decision trees are different, it is observed that some sub trees are being the same. And few Attributes have interchangeable behavior in terms of selection for splitting.

The algorithm of the FID3 paper is also implemented in Matlab environment, to make the readings comparable. Thus the Classification Performances reported in the table XII are not the same as reported in [4]. The time taken for testing the CvDT is also same as ID3 and FID3. But the time taken for

generating decision tree using CvDT is significantly less when compared to ID3 and FID3. The paired t-test on generation times reveals statistical significance, indicating that CvDT construction time is significantly lower than ID3 and FID3. In fact CvDT is outperforming the other two methods in terms of generation time when the data sets are large in size. The following figure 3 with the data sets along the X-axes with increasing sizes reveals this.

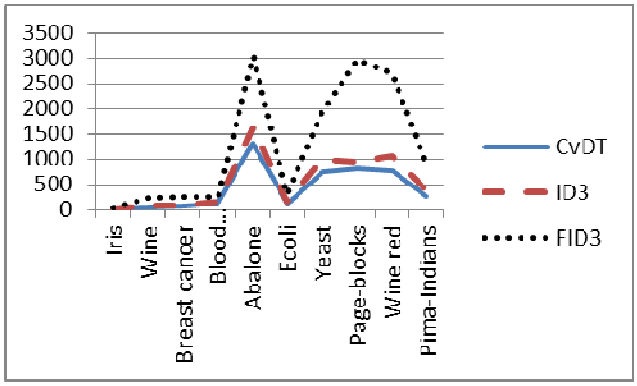

Fig. 3. Comparison of Times taken for generating(TG) the three decision trees

X. CONCLUSION

The criterion for splitting a node in a decision tree decides the efficiency of a decision tree. So far Information Gain, Gain ratio, Gini index, Chi square statistic and Kappa index are used as the splitting criteria. CvGain is proposed and demonstrated as another splitting criteria in this paper. Coefficient of Variation (Cv), which is a measure of consistency of a distribution is used to compute CvGain. It has been observed that decision tree based on CvGain has the same performance as ID3 and FID3, but at less computational cost.

ACKNOWLEDGMENT

We thank Dr. Rajeev Wankar and P.S.V.S Sai Prasad of University of Hyderabad, for their thoughtful comments and support.

REFERENCES

[1] *Blake Ian F, (1979). "An Introduction to Applied Probability", John Wiley & Sons.*

[2] *Breiman L., Friedman J., Olshen R., and Stone C (1984).. Classification and Regression Trees.Wadsworth International Group.*

[3] *Damghani K.Khalili, Taghavifard M.T., Moghaddam R. Tavakkoli (2009), Decision Making Under Uncertain and Risky situations, 2009 ERM Symposium, www.ermsymposium.org/2009/pdf/2009-damghani-decision.pdf*

[4] *Baoshi, Zheng Yongqing, Zang Shaoyu (2009), A New Decision Tree Algorithm Based on Rough Set Theory, IEEE, 2009 Asia-Pacific Conference on Information Processing.*

[5] *Fayyad U. M. and Irani K. B.(1992), The attribute selection problem in decision tree generation. In Proc. 1992 National Conference on Artificial Intelligence (AAAI'92), pages 104–110, AAAI/MITPress.*

[6] *Han Jiawei and Kamber Micheline (2006), "Data mining Concepts and Techniques", 2nd edition, Morgan Kauffman Publishers.*

[7] *Minz Sonajharia and Jain Rajni (2003), "Rough Set based Decision Tree Model for Classification", LNCS 2737 Springer.*

[8] *Mitchell Tom (1997), Machine Learning. McGraw-Hill.*

[9] *Pawlak Nsjn Zdzislaw (1991), Rough Sets – Theoretical Aspects and Reasoning about Data, Kluwer Academic Publications.*

[10] *Quinlan R. (1986),"Induction of decision trees", Machine Learning, Vol. 1, No. 1, pp.81-106.*

[11] *Quinlan R. (1993),"C4.5: Programs for Machine Learning",Morgan Kaufmann Publishers.*

[12] *Ramadevi Y, Rao C.R. (2008), Reduct based Decision Tree (RDT), IJCSES International Journal of Computer Sciences and Engineering Systems, Vol.2,No.4*

[13] *Snedecor George W., Cochran William G, (1989), "Statistical Methods", Eighth Edition, Iowa State University Press.*