Pattern Recognition 43 (2010) 2725–2731

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A new node splitting measure for decision tree construction

B. Chandra [a,*], Ravi Kothari [b], Pallath Paul [a]

[a] Department of Mathematics, Indian Institute of Technology, Delhi, India
[b] IBM Research, New Delhi, India

ARTICLE INFO

Article history:
Received 8 March 2009
Received in revised form
12 February 2010
Accepted 28 February 2010

Keywords:
Decision trees
Node splitting measure
Gini Index
Gain Ratio

ABSTRACT

A new node splitting measure termed as distinct class based splitting measure (DCSM) for decision tree induction giving importance to the number of distinct classes in a partition has been proposed in this paper. The measure is composed of the product of two terms. The first term deals with the number of distinct classes in each child partition. As the number of distinct classes in a partition increase, this first term increases and thus Purer partitions are thus preferred. The second term decreases when there are more examples of a class compared to the total number of examples in the partition. The combination thus still favors purer partition. It is shown that the DCSM satisfies two important properties that a split measure should possess viz. convexity and well-behavedness. Results obtained over several datasets indicate that decision trees induced based on the DCSM provide better classification accuracy and are more compact (have fewer nodes) than trees induced using two of the most popular node splitting measures presently in use.

1. Introduction

Top-down induction of decision trees is a powerful method of pattern classification [18]. Given a training dataset, decision trees utilize a *node splitting criteria* to partition the input space such that the training data points in each partition can be classified with lesser uncertainty. The process is recursively applied within each resulting partition not meeting a *stopping condition*.

As with other pattern classification paradigms, more complex models (larger decision trees i.e. one with more partitions or nodes) tend to produce poorer generalization performance besides being harder to humanly comprehend. The decision tree literature thus shows continuous contributions directed towards producing decision trees of smaller size.

The methods for producing smaller decision trees can be implemented during the construction of the tree (such as a new node splitting criteria or a new stopping criterion) or implemented after the construction of the tree (such as pruning). Methods in either categories are insufficient in themselves and one generally has to resort to methods to produce smaller decision trees followed by methods that prune the constructed tree in order to arrive at the smallest tree. The node splitting measure is primary amongst the techniques that can be implemented during the construction of the decision tree. Though there have been proposals for new node splitting measures, the most popular ones remain the information theoretic variants [19,20] and the Gini Index [2]. Motivated by performance and comprehensibility considerations, we propose a new node splitting measure (DCSM) in this paper. We show that DCSM is convex and well behaved. Our results over a large number of datasets indicate that decision trees constructed using DCSM are smaller and have higher classification accuracy.

We have laid out the rest of the paper as follows. In Section 2, we recall two popular node splitting measures. Our intent is not to provide a comprehensive review but to provide details on the most popular measures that are also relevant for the rest of the paper. In Section 3, we introduce the proposed node splitting measure and derive some properties of DCSM. In Section 4, we provide an algorithm to construct decision trees utilizing DCSM. In Section 5, we provide results obtained with DCSM and compare it to the results obtained from the use of the two popular node splitting measures. Our results focus on comparing the performance resulting from the node splitting measure alone; we anticipate the benefits resulting from other enhancements to benefit any existing or new node splitting measures. In Section 6, we present our conclusions.

2. Two popular node splitting measures

In this section we describe two popular split measures. Our intent is not to provide an exhaustive review but rather is to provide an overview of those measures that are required to make the paper self-contained. A more extensive though dated review appears in [22].

* Corresponding author.
E-mail addresses: bchandra104@yahoo.co.in (B. Chandra), rkothari.in@ibm.com (R. Kothari), pallathpv@yahoo.com (P. Paul).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.02.025

The decision tree is to be induced from N training examples represented as

$$(x^{(1)},y^{(1)}),(x^{(2)},y^{(2)}),\ldots,(x^{(N)},y^{(N)})$$

where $x^{(i)}$ is a vector of n attributes and $y^{(i)} \in \{\omega_1,\omega_2,\ldots,\omega_C\}$ is the class label corresponding to the input $x^{(i)}$. At a particular node, v, let there be $N^{(v)}$ training examples (for the root node, or node 0, $N^{(0)}=N$). The number of training examples at node v belonging to class ω_k is denoted by $N^{(v)}_{\omega_k}$. $\sum_k N^{(v)}_{\omega_k} = N^{(v)}$.

2.1. Gain Ratio

Entropy based node splitting criteria is based on choosing a partitioning that results in the largest decrease in entropy [18]. Consider a node u with V child nodes resulting from the partitioning induced at node u. Succinctly, the gain in information resulting from splitting the training examples based on an attribute x_j can be written as

$$Gain(x_j) = \left[\sum_{k=1}^{C} -\left(\frac{N^{(u)}_{\omega_k}}{N^{(u)}}\right)\log\left(\frac{N^{(u)}_{\omega_k}}{N^{(u)}}\right)\right] - \left[\sum_{v=1}^{V}\left(\frac{N^{(v)}}{N^{(u)}}\right)\sum_{k=1}^{C} -\left(\frac{N^{(v)}_{\omega_k}}{N^{(v)}}\right)\log\left(\frac{N^{(v)}_{\omega_k}}{N^{(v)}}\right)\right] \quad (1)$$

where the first term in the above equation is the entropy at the parent node and the second term is the weighted entropy of the child nodes. The difference represents the gain in information and the attribute that produces the largest gain in information is used for partitioning. Since Eq. (1) favors attributes with a larger number of values (large number of splits), Gain Ratio [19,20] utilizes the size of the split g to normalize the gain in information. Specifically, Gain Ratio defines the size of the split g as

$$g = \sum_{v=1}^{V}\left(\frac{N^{(v)}}{N^{(u)}}\right)\log\left(\frac{N^{(v)}}{N^{(u)}}\right) \quad (2)$$

and then using the attribute that maximizes $Gain(x_j)/g$ for splitting the node.

Variations of Gain Ratio has also been proposed in the literature. Normalized Gain [13] as a split measure has also been proposed in the literature. It has been mentioned by the authors that Normalized Gain measure preforms better than Gain Ratio only under certain assumptions. Normalized Gain measure is defined as

$$NormalizedGain(x_j) = \frac{Gain(x_j)}{\log_2 n}, \quad n \geq 2 \quad (3)$$

where n is the number of partitions created due to the split.

Average Gain proposed in [4] is also a small variation of the Gain Ratio measure. This measure aims at overcoming the drawback of Gain Ratio when the split information (denominator of Gain Ratio Measure) sometimes becomes zero or very small. In this measure the information Gain is divided by the number of values the attribute can take instead of the split information. Average Gain measure is defined as

$$AverageGain(x_j) = \frac{Gain(x_j)}{|x_j|} \quad (4)$$

The drawback of this measure is that it is not able to handle numeric attributes. Also the authors have shown that the performance of Average Gain measure is at par with that of Gain Ratio.

2.2. Gini Index

The Gini Index [2] is based on

$$Gini(x_j) = \frac{1}{N}\left[\sum_{k=1}^{C}\sum_{v=1}^{V}\frac{N^{(v)2}_{\omega_k}}{N^{(v)}} - \sum_{k=1}^{C}\frac{N^{(u)2}_{\omega_k}}{N^{(u)}}\right] \quad (5)$$

The attribute chosen is one which results in the largest decrease in "impurity" computed using Eq. (5).

3. Proposed measure—DCSM

The proposed measure (DCSM) is designed to reduce the impurity of the training patterns in each partition when it is minimized. Though the motivation is similar to that of the Gini Index the exact measure that is optimized is greatly different. As before, consider a node u with V child nodes resulting from the partitioning induced at node u.

DCSM is composed of the product of two terms. The first term $D(v)*\exp(D(v))$ deals with the number of distinct classes in each child partition. Here, $v \in \{1,2,\ldots,V\}$ and $D(v)$ denotes the number of distinct classes in partition v. As the number of distinct classes in a partition increase, this first term increases. Purer partitions are thus preferred and the relative weight given to the contribution of each partition is proportional to the fraction of the training examples that lie in that specific partition. Note that $D(v)*\exp(D(v))$ decreases much sharply than simply $\exp(D(v))$ with decreasing number of classes within each partition (decreasing impurity) though not as sharply as $\exp(D(v))^2$ (see Fig. 1). Our choice seems to provide the best dynamic range over a large number of experiments.

The second term is of the form $a^{(v)}_{\omega_k}[\exp(\delta^{(v)}(1-(a^{(v)}_{\omega_k})^2))]$ where $a^{(v)}_{\omega_k} = N^{(v)}_{\omega_k}/N^{(v)}$ and $\delta^{(v)} = D(v)/D(u)$. $\delta^{(v)}$ decreases with decrease in impurity (see Fig. 2) while $(1-(a^{(v)}_{\omega_k})^2))$ decreases when there are more examples of a class compared to the total number of examples in the partition (see Fig. 3). The combination thus still favors purer partition.

None of the existing node splitting measures includes the concept of distinct classes. The DCSM node splitting measure introduces the concept of distinct classes as given in the following equation. DCSM is evaluated for each partition and a weighted sum is taken as the measure value. The weights are determined by the proportion of data in each of the partitions $N^{(v)}/N^{(u)}$. The DCSM measure $M(x_j)$ is defined for a given attribute (feature) x_j as follows:

$$M(x_j) = \sum_{v=1}^{V}\left[\frac{N^{(v)}}{N^{(u)}} * D(v)\exp(D(v)) * \sum_{k=1}^{C}[a^{(v)}_{\omega_k} * \exp(\delta^{(v)}(1-(a^{(v)}_{\omega_k})^2))]\right] \quad (6)$$

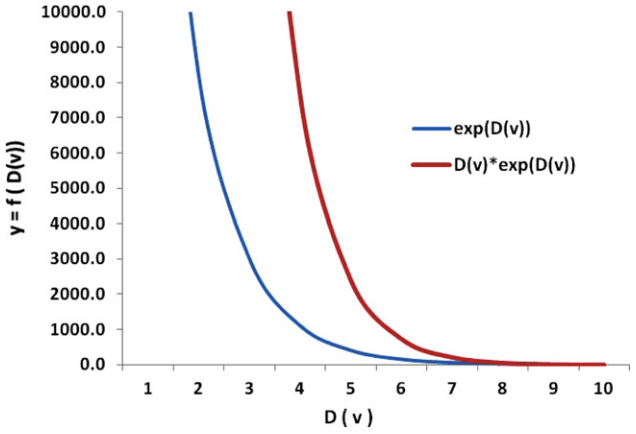

Fig. 1. Plot for $D(v)*\exp(D(v))$ and $\exp(D(v))$.

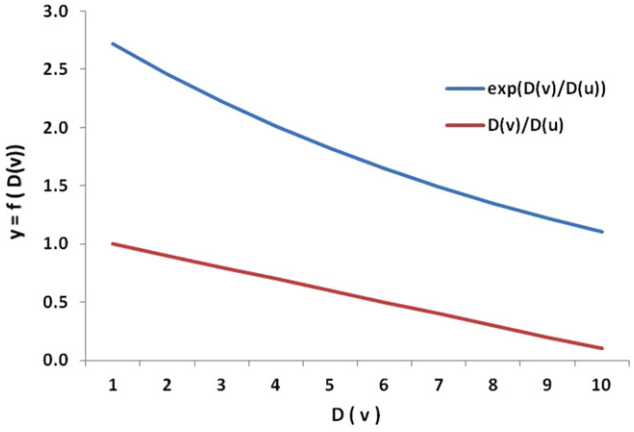

Fig. 2. Plot for $\delta^{(v)}$ and $\exp(\delta^{(v)})$.

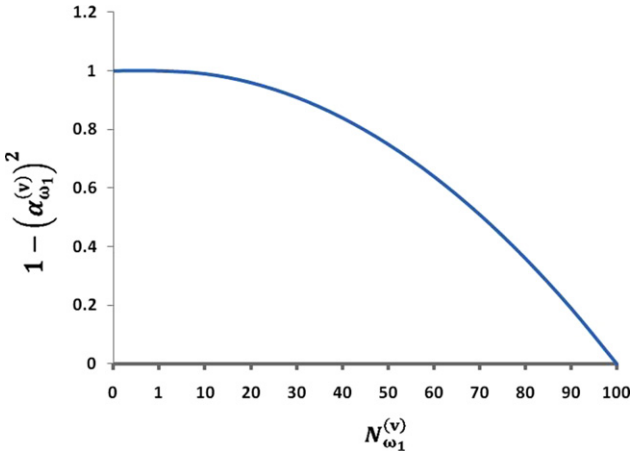

Fig. 3. Plot for $(1-(a_{\omega_k}^{(v)})^2)$.

The effect including the concept of distinct classes is that the value of the measure will increase exponentially as the number of distinct classes in the partition increases, outright rejecting such splits. This makes the measure more sensitive to the impurities present in the partition as compared to the existing measures.

Had we used only part of the split measure containing $D(v)*\exp(D(v))$ as the measure it would have not given any importance to the number of records belonging to a particular class in each partition. For example,

Case 1: Consider a split point p that partitions the dataset into two partitions in which the first partition has two records and the second has 10 records and both the partitions are have only two distinct classes.

1 2 |p| 1 2 2 1 2 1 2 1 2 2

Case 2: Following is a case of split point p in which either partition has six records or two distinct classes:

1 2 1 2 2 1 |p| 2 1 2 1 2 2

In both the cases value of $D(v)=2$ and hence the value of $D(v)*\exp(D(v))$ is same giving no importance to the number of records belonging to each class in the partition. In DCSM, the second part $\sum_{k=1}^{C}[a_{\omega_k}^{(v)} * \exp(\delta^{(v)}(1-(a_{\omega_k}^{(v)})^2))]$ handles these cases where the number of distinct classes on either side of the partition for various split points is same but the number of records belonging to a class in the partition varies. For the above example in Case 1 the value of the second term for the partitions on the left and right of split point p are 2.117 and 2.064, respectively, while for Case 2 it is 2.117 and 1.972, respectively. The value of DCSM for Case 1 is 30.637 and that for Case 2 is 30.219. As the value of DCSM is less for Case 2, this case will be given more preference as compared to Case 1. A similar example can be given out where the number of distinct classes on either of the partition varies but the number of records belonging to a class is same. In this case, the value of $D(v)*\exp(D(v))$ will be less for partition where the number of distinct classes is less while the second part of the measure will be same for all such cases. The case where the first term has minimum value will be given preference over other cases.

Thus the measure can neither be $D(v)*\exp(D(v))$ nor $\sum_{k=1}^{C}[a_{\omega_k}^{(v)} * \exp(\delta^{(v)}(1-(a_{\omega_k}^{(v)})^2))]$ alone. The combination of the two helps in picking up best split points. Two important properties which must be satisfied for a good split measure namely convexity and well behavedness hold true for DCSM split measure.

Several properties have been proposed to characterize a node splitting measure. Primary amongst those are the property of convexity and well-behavedness [1,3,5,8]. To explain these properties, consider the training examples to be sorted on the basis of the values of a given attribute (feature), say x_j. Let a *boundary* point be a value of x_j such that two consecutive examples in the list of examples sorted on the basis of x_j (one which has a value of x_j less than the boundary point and one which has a value of x_j larger than the boundary point) belong to different classes. In [8], it is shown that for binary splitting with measures that are convex downwards, the optimal split must necessarily occur at one of the boundary points. We note that convexity provides for a significant gain in efficiency since the measure need only be evaluated at the boundary points as opposed to all the possible $(N-1)$ points. The notion of well-behaved measures generalizes the notion [9] to multi-way splits [3,6,7] and convex measures are a proper subclass of well-behaved node splitting measures. More specifically, if one *bins* (discretizes) a continuous valued attribute and then merges adjacent blocks with equal relative class frequency distributions, then well-behaved measures are optimized at the so-called segment borders where a segment border is a numerical range of the attribute where the class-frequency distribution is different from the adjacent ranges.

Average Class Entropy and Information Gain have been shown to be convex [8,9] and the well-behavedness [5] property of these measures have been proved. It has been shown that Gain Ratio is not a convex function but is still well-behaved [5–7]. The Gini Index has been proved to be strictly convex [1,16], well-behaved and also cumulative [8]. In the following we show that DCSM is both convex and well-behaved. Experimental evaluations of DCSM appear in Section 5.

3.1. The proposed measure—DCSM is convex

Let the dataset of N training examples be sorted (along with the class information) by values of attribute x_j. While we could have assumed any general node, the notation becomes more cumbersome and our demonstration of convexity at the root node is without loss of generality. Let us choose $T1$ and $T2$ such that all the examples lying between $T1$ and $T2$ belong to the same class ω_C and that the class of the examples at the boundary points $T1$ and $T2$ is different from ω_C (see Fig. 4). We assume that there are $N^{(1)}$ examples to the left of a point $T1$ with $N_{\omega_k}^{(1)}$ examples of class ω_k. Similarly, we assume that there are $N^{(2)}$ examples to the right of a point $T2$ which has $N_{\omega_k}^{(2)}$ examples of class ω_k. We assume that the total number of examples between boundary points $T1$ and $T2$ is N' and the potential split point T is taken to be N'' records away from boundary point $T1$ (see Fig. 4). So, $N = N^{(1)} + N^{(2)} + N'$.

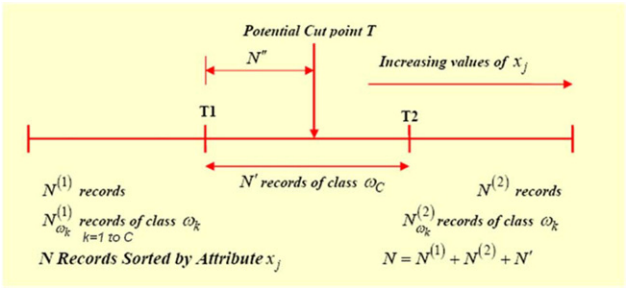

Fig. 4. Schematic associated with the proof demonstrating that DCSM is convex.

Theorem 1. *If a split point T minimizes the measure $M(x_j)$, then T is a boundary point.*

Proof. We show that the value of attribute x_j that minimizes $M(x_j)$ lies between examples of different classes in the sequence of sorted examples. The proof relies on the fact that DCSM is convex downward for $0 \le N'' \le N'$ and hence the measure has the minimum value at one of its boundary points.

Let $D(0)$ denote the number of distinct classes to which the N patterns belong. This is usually C though the use of a separate notation is to allow for the pathological instance where a given class has no training example in the training set. $D(1)$ represents the number of distinct classes in one of the partitions and $D(2)$ refers to the number of distinct classes in the other partition. For compactness, let $D(1)\exp(D(1))$ be represented by $\tilde{D}(1)$ and let $D(2)\exp(D(2))$ be represented by $\tilde{D}(2)$. Using Eq. (6) and breaking up the summation over the C classes into a summation over $(C-1)$ classes and treating the C th class separately for each partition we get

$$
\begin{aligned}
M(x_j) = {} & \left(\frac{N^{(1)}+N''}{N}\right) * \tilde{D}(1) \\
& * \left[\sum_{k=1}^{C-1}\left\{\left(\frac{N^{(1)}_{\omega_k}}{N^{(1)}+N''}\right) * \exp\left(\frac{D(1)}{D(0)}\left(1-\left(\frac{N^{(1)}_{\omega_k}}{(N^{(1)}+N'')}\right)^2\right)\right)\right\}\right] \\
& + \left(\frac{N^{(1)}+N''}{N}\right) * \tilde{D}(1) * \left[\left(\frac{N^{(1)}_{\omega_C}+N''}{N^{(1)}+N''}\right)\right. \\
& \left. * \exp\left(\frac{D(1)}{D(0)}\left(1-\left(\frac{N^{(1)}_{\omega_C}+N''}{(N^{(1)}+N'')}\right)^2\right)\right)\right] + \left(\frac{N^{(2)}+N'-N''}{N}\right) * \tilde{D}(2) \\
& * \left[\sum_{k=1}^{C-1}\left\{\left(\frac{N^{(2)}_{\omega_k}}{N^{(2)}+N'-N''}\right) * \exp\left(\frac{D(2)}{D(0)}\left(1-\left(\frac{N^{(2)}_{\omega_k}}{(N^{(2)}+N'-N'')}\right)^2\right)\right)\right\}\right] \\
& + \left(\frac{N^{(2)}+N'-N''}{N}\right) * \tilde{D}(2) * \left[\left(\frac{N^{(2)}_{\omega_C}+N'-N''}{N^{(2)}+N'-N''}\right)\right. \\
& \left. * \exp\left(\frac{D(2)}{D(0)}\left(1-\left(\frac{N^{(2)}_{\omega_C}+N'-N''}{(N^{(2)}+N'-N'')}\right)^2\right)\right)\right] \qquad (7)
\end{aligned}
$$

$$= \alpha + \beta + \gamma + \kappa \qquad (8)$$

where α, β, γ, and κ correspond to the four expressions above, respectively. Though cumbersome in terms of notation, it is straightforward to show that $d^2[M(x_j)]/d(N'')^2 < 0$ since $d^2\alpha/d(N'')^2 < 0$, $d^2\beta/d(N'')^2 < 0$, $d^2\gamma/d(N'')^2 < 0$, and $d^2\kappa/d(N'')^2 < 0$. Thus for $0 \le N'' \le N'$, $M(x_j)$ is convex downwards and hence the minimum must be at one of the extremes of the interval [14], i.e. at $N'' = 0$ or at $N'' = N'$. Hence, $M(x_j)$ is convex downward. □

Thus, the DCSM node splitting measures' minimum values for binary partitions can only occur at boundary points. "But convexity does not ensure the same for multi-way partitioning. The well-behavedness property of a split measure ensures that the optimal multi-way splits are also on boundary points. The well-behavedness of an evaluation function is a property that guarantees the optimal multi-way partitions for numerical attribute value ranges. Well-behavedness reduces the number of candidate cut points that need to be examined in multi-splitting numerical attributes" [7].

3.2. The proposed measure—DCSM is well-behaved

We show that DCSM splits the training examples at "segment borders" [7] thereby excluding all cut points that separate partitions of identical relative class frequency distributions i.e. changes in class distribution, rather than relative impurities of the partitions define the potential locations of the optimal split point.

Theorem 2. *Partitions induced by $M(x_j)$ are on segment borders.*

Proof. We consider the V-ary partition $\uplus_{v=1}^{V} N^{(v)}$ of the N training examples, where subsets $N^{(h)}$ and $N^{(h+1)}$ are composed of the set $p \cup q \cup r$. Training examples in the three subsets p, q, and r are given by $N^{(p)} = \sum_{k=1}^{C} N^{(p)}_{\omega_k}$, $N^{(q)} = \sum_{k=1}^{C} N^{(q)}_{\omega_k}$ and $N^{(r)} = \sum_{k=1}^{C} N^{(r)}_{\omega_k}$, respectively, where as before, $N^{(p)}$ represents the number of training examples in the subset p and $N^{(p)}_{\omega_k}$ represents the number of training examples of class ω_k in subset p. The notation is similar for the subsets q and r. We define $\alpha_j = (N^{(q)}_{\omega_k}/N^{(q)}) \in [0,1]$ and l to be an integer, $0 \le l \le N^{(q)}$. We assume the splitting of the set q so that the l examples belong to $N^{(h)}$ and $(N^{(q)}-l)$ examples belong to $N^{(h+1)}$ resulting in identical class frequency distributions for both subsets of q regardless of the value of l (see Fig. 5). In other words, $N^{(l)}_{\omega_j} = \alpha_j \cdot l, \forall j, l$.

The proof is based on showing that the split point chosen between $N^{(h)}$ and $N^{(h+1)}$ is on the segment border when a node is partitioned into multiple partitions. The placement of the split point between $N^{(h)}$ and $N^{(h+1)}$ does not affect the remaining partitions in the entire dataset of size N. Thus, the remaining partitions do not contribute any thing as the proof involves twice differentiation.

Let $L(l)$ denote the value of $\sum_{i=1}^{h} M_{N^{(i)}}(x_j)$ when $N^{(h)}$ contains p examples and the first l examples from q, and $R(l)$ denote the value of $\sum_{i=h+1}^{V} M_{N^{(i)}}(x_j)$ when $N^{(h+1)}$ contains r examples and the last $(N^{(q)}-l)$ examples from q. Now,

$$
\begin{aligned}
L(l) &= \sum_{i=1}^{h-1} M_{N^{(i)}}(x_j) + M_{N^{(h)}}(x_j) \\
R(l) &= M_{N^{(h+1)}}(x_j) + \sum_{i=h+2}^{V} M_{N^{(i)}}(x_j) \qquad (9)
\end{aligned}
$$

Since the first term of $L(l)$ and the last term of $R(l)$ are independent of the placement of the h th split point, it differentiates to 0.

Using the definition of the measure given in Eqs. (6) and (9), it is straightforward to show that $d^2L(l)/dl^2 < 0$ and $d^2R(l)/dl^2 < 0$ as

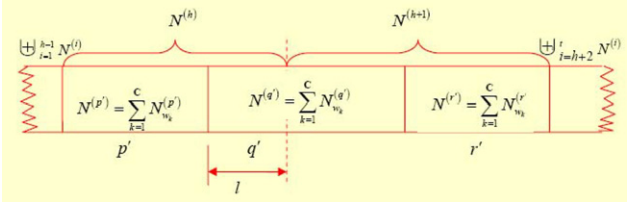

Fig. 5. Schematic associated with the proof demonstrating that DCSM is well-behaved.

Table 1
List of datasets used.

No.	Dataset name	No. of attributes (n)	No of classes (C)
1	Balanced Scale	4	3
2	Liver	6	2
3	Wisconsin BC	9	2
4	Echocardiogram	9	2
5	Wine	13	3
6	Mushroom	21	2
7	Ionosphere	34	2
8	Lung Cancer	54	3
9	Image	19	7
10	Glass	9	6
11	Vehicle Silhoutte	18	4
12	Voting	16	2
13	Heart Statlog	13	2
14	Lymphograph	18	4
15	Waveform	21	3
16	OptoDigits	64	10
17	Pen Digits	16	10
18	Madelon	500	2
19	Dermatology	34	6
20	Twonorm	20	2
21	Sonar	60	2
22	Page Block	10	5

well. Hence, DCSM measure does not obtain its minimum value within the segment q. □

4. Results

4.1. Results: un-pruned decision trees

To assess the relative performance of DCSM measure, we use 22 different datasets (see Table 1) from the UCI machine-learning repository (http://www.ics.uci.edu/~mlearn/MLRepository.html) and compare the resulting decision tree with ones obtained using Gini Index (as used in SLIQ [15]) and Gain Ratio. We report the classification accuracy of the decision tree, the size of the decision tree, and the time taken to construct the decision tree using each of the methods. Our results are based on the average of 10-fold cross-validation runs.

Table 2 shows the classification performance of the decision trees constructed using the different node splitting measures. The numbers in the table are reported as the average accuracy over the 10 cross-validation runs ± the standard deviation of the accuracy obtained in those runs. The results show that DCSM measure has the highest classification accuracy for 17 of the 22 datasets. Indeed, with the exception of the Ionosphere dataset, DCSM provides better average classification accuracy over all the datasets.

The performance of the decision trees built using DCSM in comparison to that built using Gain Ratio and Gini Index is evaluated using the method proposed in [12]. Demsar suggests the use of Iman's F statistic [21] which uses Friedman's χ_F^2 statistic [10,11] for comparing multiple classifiers over several datasets. If the null hypothesis (all the algorithms are equivalent) is rejected then Demsar proposes to use the post-hoc test: Nemenyi test [17] to find which classifier performance significantly better than the others. According to the Nemenyi test the performance of two classifiers is significantly different if the corresponding average ranks differ by at least the critical difference.

The average rank based on classification accuracy for Gini Index, Gain Ratio and DCSM based un-pruned decision trees are 2.20, 2.34 and 1.45, respectively. The average rank suggest that DCSM is the best performing measure compared to Gain Ratio and Gini Index. The computation of Friedman's χ_F^2 statistic is shown below:

$$\chi_F^2 = \frac{12\times 22}{3\times 4}\left[(2.2046)^2+(2.341)^2+(1.455)^2-\frac{3\times 4^2}{4}\right] = 10.02 \tag{10}$$

Iman's F statistic is given below:

$$F_F = \frac{(22-1)\times(10.02)}{22\times(3-1)-10.02} = 6.20 \tag{11}$$

Critical value of $F(2,63)$ for $\alpha = 0.05$ is 3.14. Since $F_F > F_{\alpha=0.05}(2,63)$, the null-hypothesis is rejected. We proceed with a post-hoc Nemenyi test [17] to find which measure gives better results. The critical difference CD is given below:

$$CD = 2.343\sqrt[2]{\frac{3(3+1)}{6\times 22}} = 0.71 \tag{12}$$

The difference between average rank of DCSM and Gini index is 0.75 and that between DCSM and Gain Ratio is 0.89. Since both the differences are greater than CD, the performance of DCSM is significantly better than both Gain Ratio and Gini Index.

Table 3 compares the size of the decision tree (number of nodes) constructed using the different node splitting measures. The number of nodes resulting from the use of DCSM is less in 18 of the 22 datasets compared to that of Gini Index and Gain Ratio.

Finally, Table 4 compares the time taken to construct the decision trees using the different node splitting measures. We observe that the time taken to construct the decision tree using any of the node splitting measures are comparable. Alternatively, DCSM provides far superior classification accuracy without requiring any additional time to construct the decision tree.

Table 2
% Standard errors of decision trees constructed using different node splitting measures.

No.	Dataset name	Gini Index	Gain Ratio	DCSM
1	Balanced Scale	22.26 ± 7.75	25.32 ± 15.34	22.42 ± 5.61
2	Liver	42.86 ± 12.70	44.29 ± 9.54	40.48 ± 12.35
3	Wisconsin BC	6.67 ± 4.84	5.51 ± 4.57	4.64 ± 4.62
4	Echocardiogram	16.00 ± 13.50	15.00 ± 12.69	15.00 ± 11.79
5	Wine	11.67 ± 8.05	3.33 ± 3.88	5.56 ± 3.70
6	Mushroom	1.87 ± 3.76	0.44 ± 1.40	1.62 ± 3.84
7	Ionosphere	11.43 ± 7.50	8.00 ± 4.22	12.29 ± 9.62
8	Lung Cancer	42.5 ± 20.58	57.5 ± 16.87	42.5 ± 16.87
9	Image	6.00 ± 3.14	8.79 ± 10.12	5.28 ± 1.56
10	Glass	35.00 ± 16.04	39.52 ± 11.01	35.24 ± 12.34
11	Vehicle Silhoutte	26.76 ± 6.66	28.94 ± 7.32	27.88 ± 5.76
12	Voting	6.52 ± 4.70	6.52 ± 6.23	5.22 ± 4.49
13	Heart Statlog	25.67 ± 3.87	27.33 ± 5.84	25.00 ± 5.27
14	Lymphograph	21.43 ± 13.88	22.86 ± 12.05	20.00 ± 11.57
15	Waveform	28.83 ± 7.90	29.00 ± 6.81	27.17 ± 8.50
16	OptoDigits	11.39 ± 2.73	11.58 ± 2.20	9.18 ± 0.64
17	Pen Digits	3.91 ± 0.84	3.79 ± 0.65	3.45 ± 0.81
18	Madelon	37.40 ± 9.94	50.00 ± 6.04	37.6 ± 8.83
19	Dermatology	6.57 ± 4.87	5.43 ± 4.94	4.86 ± 4.48
20	Twonorm	15.91 ± 1.11	16.05 ± 1.15	13.74 ± 1.57
21	Sonar	44.00 ± 15.42	48.00 ± 17.19	38.00 ± 14.94
22	Page Block	4.64 ± 2.22	4.61 ± 1.93	4.41 ± 2.41

Numbers are based on 10-fold cross-validation and reported as the average %standard error of those runs ± the standard deviation of the standard error obtained in those runs.

4.2. Results: pruned decision trees

To evaluate the performance of the DCSM node splitting measure after pruning, MDL pruning [15] was employed on decision trees built using all the three measure. The classification accuracy and the size of the decision tree is compared with that obtained using pruned decision trees constructed using Gain Ratio

Table 3
Size of the decision trees (number of nodes) constructed using different node splitting measures.

No.	Dataset name	Gini Index	Gain Ratio	DCSM
1	Balanced Scale	129.00	101.10	115.10
2	Liver	42.80	113.90	42.50
3	Wisconsin BC	28.20	30.20	24.60
4	Echocardiogram	19.00	20.60	13.80
5	Wine	10.80	8.90	7.40
6	Mushroom	17.40	30.60	24.00
7	Ionosphere	22.70	21.20	20.20
8	Lung Cancer	10.00	10.10	9.00
9	Image	64.90	74.70	56.80
10	Glass	43.80	48.50	42.30
11	Vehicle Silhoutte	120.30	151.60	118.70
12	Voting	13.60	12.90	13.10
13	Heart Statlog	39.00	46.90	39.70
14	Lymphograph	26.00	31.10	25.50
15	Waveform	65.30	93.30	58.20
16	OptoDigits	212.80	252.70	202.70
17	Pen Digits	214.20	240.20	181.60
18	Madelon	53.30	134.20	48.50
19	Dermatology	16.70	16.90	19.70
20	Twonorm	418.80	601.00	347.50
21	Sonar	19.00	33.00	16.60
22	Page Block	146.80	164.20	145.90

Numbers are based on the average of the number of nodes resulting in 10-fold cross-validation runs.

Table 4
Time taken to construct the decision trees using different node splitting measures.

No.	Dataset name	Gini Index	Gain Ratio	DCSM
1	Balanced Scale	0.553	0.219	0.229
2	Liver	0.245	0.701	0.265
3	Wisconsin BC	0.727	0.706	0.564
4	Echocardiogram	0.312	0.398	0.253
5	Wine	0.264	0.180	0.160
6	Mushroom	61.842	116.653	78.552
7	Ionosphere	5.626	4.494	4.033
8	Lung Cancer	0.074	0.054	0.050
9	Image	17.019	17.063	21.030
10	Glass	0.514	0.417	0.365
11	Vehicle Silhoutte	8.908	5.479	3.520
12	Voting	0.731	0.298	0.309
13	Heart Statlog	0.888	0.527	0.462
14	Lymphograph	1.684	0.466	0.374
15	Waveform	4.103	5.476	3.653
16	OptoDigits	184.103	227.900	172.498
17	Pen Digits	88.817	146.534	92.626
18	Madelon	317.486	1029.522	324.443
19	Dermatology	0.845	0.891	0.946
20	Twonorm	407.580	1330.000	418.200
21	Sonar	1.449	2.865	1.551
22	Page Block	115.715	165.870	58.662

Numbers are reported in seconds.

and Gini Index. The average accuracy (after 10-cross-validation) obtained for the pruned decision trees is given in Table 5. DCSM performs better than Gain Ratio and Gini Index in out of 22 datasets.

The performance of the pruned decision trees built using DCSM, Gain Ratio and Gini Index as node splitting measure is evaluated using the method proposed in [12]. The average rank of Gini Index, Gain Ratio and DCSM based un-pruned decision trees are 2.34, 2.20 and 1.41, respectively. The average rank suggest that DCSM is the best performing measure compared to Gain Ratio and Gini Index. The computation of χ_F^2 is shown below:

$$\chi_F^2 = \frac{12\times 22}{3\times 4}\left[(2.341)^2+(2.2045)^2+(1.409)^2-\frac{3\times 4^2}{4}\right] = 7.19 \tag{13}$$

Table 5
% Standard errors of decision trees constructed using different node splitting measures.

No.	Dataset name	Gini Index	Gain Ratio	DCSM
1	Balanced Scale	32.58 ± 8.08	34.35 ± 12.36	31.84 ± 9.50
2	Liver	44.29 ± 8.99	35.71 ± 11.93	37.33 ± 6.90
3	Wisconsin BC	6.81 ± 4.21	5.51 ± 4.26	4.49 ± 3.24
4	Echocardiogram	16.00 ± 13.50	11.00 ± 11.01	11.00 ± 12.87
5	Wine	11.67 ± 8.05	3.33 ± 3.88	2.56 ± 3.70
6	Mushroom	1.87 ± 3.76	0.44 ± 1.40	0.62 ± 3.84
7	Ionosphere	11.71 ± 7.67	6.57 ± 4.68	9.29 ± 9.62
8	Lung Cancer	50.00 ± 23.57	60.00 ± 17.48	42.5 ± 16.87
9	Image	6.15 ± 3.05	8.74 ± 10.15	5.37 ± 1.58
10	Glass	36.19 ± 15.75	35.71 ± 16.69	34.76 ± 13.48
11	Vehicle Silhoutte	30.47 ± 5.75	38.24 ± 13.17	25.76 ± 6.41
12	Voting	3.48 ± 3.43	3.48 ± 3.43	3.48 ± 3.43
13	Heart Statlog	24.33 ± 4.98	27.33 ± 6.05	25.00 ± 6.71
14	Lymphograph	33.57 ± 9.55	27.14 ± 13.80	25.71 ± 10.75
15	Waveform	29.00 ± 7.71	26.71 ± 6.53	27.33 ± 8.47
16	OptoDigits	11.37 ± 2.69	11.55 ± 2.15	9.43 ± 1.13
17	Pen Digits	3.89 ± 0.86	4.09 ± 0.79	3.43 ± 0.80
18	Madelon	37.40 ± 9.94	54.00 ± 7.54	37.40 ± 8.90
19	Dermatology	6.57 ± 4.87	5.43 ± 4.94	5.29 ± 4.63
20	Twonorm	15.88 ± 1.14	16.74 ± 1.57	13.74 ± 1.57
21	Sonar	44.00 ± 15.42	57.50 ± 8.58	38.00 ± 14.94
22	Page Block	4.31 ± 1.98	3.62 ± 1.85	3.80 ± 2.08

Numbers are based on 10-fold cross-validation and reported as the average standard error of those runs ± the standard deviation of the standard error obtained in those runs after MDL pruning.

Table 6
Size of the decision trees (number of nodes) constructed using different node splitting measures.

No.	Dataset name	Gini Index	Gain Ratio	DCSM
1	Balanced Scale	36.00	36.20	22.30
2	Liver	5.10	2.00	3.20
3	Wisconsin BC	14.30	14.90	13.50
4	Echocardiogram	6.00	3.40	3.80
5	Wine	8.10	7.20	6.40
6	Mushroom	16.40	14.30	11.00
7	Ionosphere	18.80	12.70	13.10
8	Lung Cancer	5.80	5.70	4.30
9	Image	60.60	45.00	44.10
10	Glass	22.70	18.80	13.20
11	Vehicle Silhoutte	64.00	25.20	26.10
12	Voting	3.20	3.20	3.20
13	Heart Statlog	16.00	2.70	3.20
14	Lymphograph	13.30	14.60	13.90
15	Waveform	56.70	33.70	37.00
16	OptoDigits	211.00	249.40	201.70
17	Pen Digits	208.80	212.50	175.20
18	Madelon	52.40	50.60	47.50
19	Dermatology	14.40	14.00	14.50
20	Twonorm	389.20	746.3	346.30
21	Sonar	17.50	15.70	15.60
22	Page Block	64.40	65.10	67.70

Numbers are based on the average of the number of nodes resulting in 10-fold cross-validation runs after MDL pruning.

Iman's F statistic is given below:

$$F_F = \frac{(22-1)\times(7.19)}{22\times(3-1)-7.19} = 4.08 \tag{14}$$

Critical value of $F(2,63)$ for $\alpha = 0.05$ is 3.14. Since $F_F > F_{\alpha=0.05}(2,63)$, the null-hypothesis is rejected. The post-hoc Nemenyi test [17] is carried out to find classifier built using which measure gave better results. The critical difference 0.71.

The difference between average rank of DCSM and Gini Index is 0.93 and that between DCSM and Gain Ratio is 0.79. Since both

the differences are greater than CD, the performance of the DCSM is significantly better than both Gain Ratio and Gini Index.

The size of the pruned decision tree build using DCSM is least in 13 out of 22 datasets compared to that of Gain Ratio and Gini Index. The average size (after 10-fold cross validation) of the pruned decision tree build using Gini Index, Gain Ratio and DCSM is given in Table 6.

5. Conclusion

Node splitting measures are primary amongst the techniques that can be implemented during the construction of decision trees and represent one aspect of a multi-part approach for producing compact decision trees with improved generalization abilities. In this paper, we presented a new node splitting measure. We showed that DCSM is convex and well-behaved. Our results provide compelling evidence that decision trees produced using DCSM are more compact and provide better classification accuracy than trees constructed using two of the presently popular node splitting measures (the Gini Index and the Gain Ratio). DCSM measure also enjoys the benefits of pruning (an approach to producing compact trees with better classification accuracy).

References

[1] L. Breiman, Some properties of splitting criteria, Machine Learning 24 (1996) 41–47.
[2] L. Breiman, J. Friedman, R. Olsen, C. Stone, Classification and Regression Trees, Wadsworth International, 1984.
[3] C. Codrington, C.E. Brodley, On the qualitative behavior of impurity based splitting rules I: the minima-free property, Technical Report, Purdue University, 1997.
[4] W. Dianhong, J. Liangxiao, An improved attribute selection measure for decision tree induction, in: Fourth International Conference Proceedings on Fuzzy Systems and Knowledge Discovery—FSDK 2007, vol. 4, IEEE CS, 2007, pp. 654–658.
[5] T. Elomma, J. Rousu, On the well-behavedness of important attribute evaluation functions, in: Proceedings of Sixth Scandinavian Conference on Artificial Intelligence, IOS Press, 1997, pp. 95–106.
[6] T. Elomma, J. Rousu, General and efficient multisplitting of numerical attributes, Machine Learning 1 (1999) 1–49.
[7] T. Elomma, J. Rousu, On splitting properties of common attribute evaluation functions, Report C-2000-1, Department of Computer Science, University of Helsinki, 2000.
[8] U. Fayyad, K.B. Irani, On the handling of continuous-valued attributes in decision tree generation, Machine Learning 8 (1992) 87–102.
[9] U. Fayyad, K.B. Irani, Multi-interval discretization of continuous valued attributes for classification learning, in: The Thirteenth International Joint Conference on Artificial Intelligence Conference Proceedings, Morgan Kaufmann, 1993, pp. 1022–1027.
[10] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analyis of variance, Journal of the American Statistical Association 32 (1937) 675–701.
[11] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, Annals of Mathematical Statistics 11 (1940) 86–92.
[12] J. Demsar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.
[13] B.H. Jun, C.S. Kim, J. Kim, A new criterion in selection and discretization of attributes for the generation of decision trees, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (12) (1997) 1371–1375.
[14] D. Luenberger, Introduction to Linear and Nonlinear Programming, Addison-Wesley, Reading, 1973.
[15] M. Mehta, R. Agrawal, J. Riassnen, Sliq: a fast scalable classifier for data mining, in: Extending Database Technology, Springer, 1996, pp. 18–32.
[16] Y. Morimoto, Algorithms for finding attribute value group for binary segmentation of categorical databases, IEEE Transactions on Knowledge and Data Engineering 14 (6) (2002) 1269–1279.
[17] P.B. Nemenyi, Distribution—free multiple comparisons, Ph.D. Thesis, Princeton University, 1963.
[18] J. Quinlan, Induction of decision trees, Machine Learning (1986) 81–106.
[19] J. Quinlan, C4.5: Programs for Machine Learning, Springer 16 (3) (1993) 235–240.
[20] J. Quinlan, Improved use of continuous attributes in c4.5, Journal of Artificial Intelligence 4 (1996) 77–90.
[21] R.L. Iman, J.M. Davenport, Approximations of the critical region of the Friedman statistic, Communications in Statistics (1980) 571–595.
[22] S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE Transactions on Systems Man and Cybernetics 21 (1991) 660–674.

About the Author—B. CHANDRA is with the Computing Group, Department of Mathematics, Indian Institute of Technology, Delhi, India, where she is currently a Professor and was the Department Chair from August 2004 to August 2007. She has been a Visiting Professor with the Graduate School of Business, University of Pittsburgh, Pittsburgh, and Penn State University, University Park. She has also been a Visiting Scientist with the Institut National de Recherche en Informatique et en Automatique, France. She has been the Chairman in various sessions on neural networks and machine learning at international conferences held at Hawaii, Washington, DC in the U.S., Bangor, U.K., Montreal, Canada, Singapore, Jeon-Buk, South Korea, and at Bangkok, Thailand. She has been invited to deliver invited lectures at various universities in the U.S., viz. University of Pittsburgh, Penn State University, University of Eastern Illinois, University of Hawaii, University of Texas at Dallas, University of Orleans, and Virginia Tech, and at other institutions like the National University of Singapore, Bangor University, U.K., and Ecole de Mines Paris. She has been the Principal Investigator of many sponsored research and consultancy research projects in the field of neural networks and machine learning. She is also actively involved in teaching and curriculum development for the Graduate Program in Computer Applications at the Indian Institute of Technology. She has authored a number of research papers published in reputed international journals in the area of neural networks, classification, and clustering. She has also authored three books.

About the Author—RAVI KOTHARI started his professional career as an Assistant Professor in the Department of Electrical and Computer Engineering and Computer Science (ECECS) at the University of Cincinnati, Cincinnati, OH (USA) where he later became a tenured Associate Professor and Director of the Artificial Neural Systems Laboratory. His work has centered around pattern recognition, machine learning, self-organization, mathematical modeling, adaptation, and application of these technologies to various areas. Since 2002, he has been with IBM-India Research Laboratory, New Delhi, India.

Dr. Kothari has served as an Associate Editor of the IEEE Transactions on Neural Networks, IEEE Transactions on Knowledge and Data Engineering, Pattern Analysis and Applications (Springer) as well as on the program committees of various conferences. He is an IEEE Distinguished Visitor and is a member of the IBM Academy of Technology.

About the Author—PALLATH PAUL VARGHESE did his PhD from the Indian Institute of Technology, Delhi, in the area of Machine Learning Algorithms and its applications. He is a gold medalist in M.Tech. in Computer Applications from IIT Delhi. His research interests include, data mining, clustering and artificial neural networks.