December 2002

SPLITTING METHODS FOR DECISION TREE INDUCTION: A COMPARISON OF TWO FAMILIES

Kweku-Muata Osei-Bryson
Virginia Commonwealth University

Kendall Giles
Virginia Commonwealth University

SPLITTING METHODS FOR DECISION TREE INDUCTION: A COMPARISON OF TWO FAMILIES

Kweku-Muata Osei-Bryson and Kendall Giles
Department of Information Systems and
The Information Systems Research Institute
Virginia Commonwealth University
Kweku.Muata@isy.vcu.edu kgiles@acm.org

Abstract

Decision tree (DT) induction is among the more popular of the data mining techniques. An important component of DT induction algorithms is the splitting method, with the most commonly used method being based on the conditional entropy family. However, it is well known that there is no single splitting method that will give the best performance for all problem instances. In this paper we explore the relative performance Conditional Entropy family and another family that is based on the Class-Attribute Mutual Information (CAMI) measure. Our results suggest that while some datasets are insensitive to the choice of splitting methods, other datasets are very sensitive to the choice of splitting methods. For example, some of the CAMI family methods may be more appropriate than GainRatio (GR) for datasets where all non-class attributes are nominal; some of the CAMI methods perform as well as GR for datasets where all the non-class attributes are either integer or continuous. Given the fact that it is never known beforehand which splitting method will lead to the best DT for the given dataset, and given the relatively good performance of the CAMI methods, it seems appropriate to suggest that splitting methods from the CAMI family should be included in data mining toolsets.

Keywords: Decision trees, entropy, splitting methods, classification, machine learning

Introduction

Over the past two decades there has been an increased interest in the use of data mining techniques to address problems in various fields. Among the more popular of these tasks is classification, and for this task various classification algorithms have been proposed, such as decision trees, neural networks, linear discriminant analysis, nonparametric methods, and statistical methods (e.g. Bradley et. al., 1999; Wu and Urpani, 1999; Cheesean and Stutz, 1996; Ching et. al., 1995; Safavian and Landgrebe, 1991; Quinlan, 1986). In this study we concentrate on decision tree (DT) induction algorithms, and in particular those that use entropy-based splitting methods.

A splitting method is the component of the DT induction algorithm that determines both the attribute that is selected for a given node of the DT and also the partitioning of the values of the selected attribute into mutually exclusive subsets such that each subset uniquely applies to one of the branches that emanate from the given node. While the most commonly used splitting methods are based on the conditional entropy (CE) family (e.g. Quinlan's C4.5 family of decision tree induction algorithms), it is well known that there is no single splitting method that will give the best performance for all problem instances. A question that interested us is how well would other families of entropy measures perform compared to the Conditional Entropy family of entropy measures. With this in mind we chose to compare the performance of the Conditional Entropy (CE) family and another family that is based on a measure called Class-Attribute Mutual Information (CAMI) that was proposed by Ching et. al. (1995). Bryson (2000) had previously identified some of the conceptual links between both families, and developed a new splitting method, EffCAMI. Our computational exploration had the objective of testing the measures using a wide-variety of datasets from different problem domains and with different data characteristics, and directly comparing the classification accuracy results from both families.

This paper is organized as follows: in Section 2 we present an overview of the entropy-based splitting methods as families, and introduce a new measure, called EffCAMI. In Section 3 we present the results of our experiments that compare the six entropy measures using twenty-five (25) datasets. Section 4 presents our analysis and conclusions.

Overview of the Two Entropy-Based Families

Definition of Terms

Let n be the total number of examples in the dataset; $n_{j\bullet}$ be the total number of examples in interval j of the given attribute; $n\bullet s$ be the total number of examples in class s; $n_{j\wedge s}$ be the total number of examples in interval j and class s; $p_{j\bullet} = (n_{j\bullet}/n)$ be the estimated probability of being in interval j; $p\bullet s = (n_{\bullet s}/n)$ be the estimated probability of being in class s; $p_{j\wedge s} = (n_{j\wedge s}/n)$ be the estimated probability of being in interval j and class s; $p_{s|j} = (n_{j\wedge s}/n_{j\bullet}) = (p_{j\wedge s}/p_{j\bullet})$ be the conditional probability of an example being in class s given that it is in interval j; S be the set of classes.

Conditional Entropy family

The **Information Gain** measure, (e.g. Quinlan, 1993) is based on maximizing the "gain" in selecting a particular attribute for branching when creating the tree. In the equation below, the first term represents the entropy considering all the attributes together and the second term represents the entropy between the class and attribute. The attribute that results in the largest difference represents the best attribute on which to split. In other words, the goal is to attempt to measure the increase in the information that results from the discretization of an attribute that takes its values from an ordered domain.

The *Information Gain (IG)* for a discretization Γ_g of an attribute into g intervals is defined as:

$$IG(g) = -\sum_{s\in S} p_{\bullet s}\log_2(p_{\bullet s}) \; - \; \sum_{j\in J_{\Gamma g}} p_{j\bullet}(-\sum_{s\in S} p_{s|j}\log_2(p_{s|j})),$$

where $J_{\Gamma g}$ is the index set of the intervals that are included in a particular discretization Γ_g that consists of $g = |\Gamma_g|$ intervals. An alternate approach involves the maximization of the *Gain Ratio GR(g)* = IG(g)/SI(g), where $SI(g) = \sum_{j\in J_{\Gamma g}} -p_{j\bullet}\log_2(p_{j\bullet})$ is called the *Split Information* for the partition Γ_g with g intervals, by Quinlan (1993). However, for some datasets the Gain Ratio measure "overcompensates" [18] and so must be moderated by choosing the attribute with the Max Gain Ratio and an above average Information Gain. An examination of the Information Gain formula shows a component (i.e. the Unconditional or A Priori Entropy) that is the same for all attributes and all values of g, and a second major component that is dependent on the relevant attribute and also on the value of g. This second component is called **Conditional Entropy** $H(Class|\, a_i)$ for attribute a_i, and is defined as $H(Class|a_i) = -\sum_{j\in J_{\Gamma gi}} p_{j\bullet}\sum_{s\in S} p_{s|j}\log_2(p_{s|j})$. With this measure the objective is to minimize the conditional entropy between class and attribute. The reasoning is that the attribute that gives the lowest conditional entropy value with a particular class represents the best attribute to split on. The reader may have observed that maximizing the Information Gain is equivalent to minimizing the Conditional Entropy.

Class-Attribute Mutual Information (CAMI) Entropy Family

The **Class-Attribute Mutual Information (CAMI)** measure proposed by Ching et. al. (1995) is a non-decreasing function of the number of intervals *g*, where $g > 1$. The CAMI measure is defined as $CAMI(g) = \sum_{j\in J_{\Gamma g}} \sum_{s\in S} p_{j\wedge s}\log_2(p_{j\wedge s}/p_{j\bullet}p_{\bullet s})$, where Γ_g represents a discretization with *g* intervals. Bryson (2000) showed that for $g \le |S|$, SupCAMI(g), the maximum possible value of CAMI(g), is equal to $\sum_{j\in J_{\Gamma g}} -p_{j\bullet}\log_2(p_{j\bullet})$, which is the same as Quinlan's Split Information Measure (Quinlan, 1993) for a discretization of the attribute into g intervals based on partition Γ_g.; and for $g \ge |S|$, SupCAMI(g) is equal to $\sum_{s\hat{I}S} -p\bullet s\log_2(p\bullet s)$. Because CAMI(g) is a non-decreasing function of the number of intervals g ($g > 1$), Ching et. al. defined a second measure, the **Class-Attribute Interdependence Redundancy** (CAIR) such that CAIR(g) = CAMI(g)/Max{$\log_2(|S|)$, $\log_2(g)$}, and proposed that the attribute discretization problem could be solved by finding the value of g that maximized CAIR(g). They based this approach on a claim that Max{$\log_2(|S|)$, $\log_2(g)$} was the maximum value of CAMI(g), and so CAIR(g) = 1 if there is perfect attribute/class interdependency, and CAIR(g) = 0 if there is absolutely no attribute/class interdependency. Bryson (2000) proved that Ching et. al.'s assertion, that the maximum possible value of CAMI(g) is equal to Max{$\log_2(|S|)$, $\log_2(g)$}, is not correct. A more plausible rationale for using CAIR(g) is that it provides a trade-off between the improvement in the class-attribute mutual

information and the cost of the number of intervals. Bryson (2000) proposed a new measure for this family: **EffCAMI** = Max{CAMI(g)/SupCAMI(g), g = 2,…, g_{prac}}, where g_{prac} is the maximum number of intervals that are appropriate for the given decision tree induction algorithm, and EFFCAMI could be considered as a measure of the relative strength of the class-attribute interdependence provided by partition Γ_g of the given attribute.

Experimental Exploration

Software Environment

As mentioned previously we wanted to explore the relative performance of our entropy measures from both families with regard to the generation of decision trees. We thus implemented these entropy measures (i.e. Gain Ratio, Conditional Entropy, CAMI, EffCAMI, and CAIR) in the Weka library implementation of the well-known C4.5 algorithm, complete with pruning and statistic calculation (http://www.cs.waikato.ac.nz/~ml/index.html). The C4.5 algorithm uses Information Gain and Gain Ratio as the decision criteria for choosing an appropriate attribute for branching. In order to test Conditional Entropy, CAMI, CAIR, and EffCAMI, we wrote our own Java programs and classes to use the C4.5 algorithm structure in the Weka Java library, and substituted the other entropy measures in place of Information Gain and Gain Ratio. Table 1 contains a summary of the entropy measures we used and the decision rule for each measure.

Table 1. Induction Algorithm Decision Rules for Selecting The Best Attribute

Entropy Measure	Decision Rule
GainRatio	For those attributes whose InfoGain > Average(InfoGain), select the attribute that provides Max(GainRatio).
Conditional Entropy	Select the attribute that provides Min(ConditionalEntropy).
CAMI	Select the attribute that provides Max(CAMI).
CAIR	For those attributes whose CAMI > Average(CAMI), select the attribute that provides Max(CAIR).
EffCAMI_0	For those attributes whose CAMI > Average(CAMI), select the attribute that provides Max(EffCAMI).
EffCAMI_1	For those attributes whose CAMI > (Average(CAMI) - StandardDeviation(CAMI)), select the attribute that provides Max(EffCAMI).

Test Problems

We ran our entropy measures on 25 publicly available benchmark data mining datasets that we obtained from the UCI Irvine machine library: IRIS, Breast Cancer, Car, Credit Approval, Abalone, Glass, Soybean, Page Blocks, Mushroom, Waveform, Wine, Yeast, Zoo, Pima Indians, Nursery, Audiology, Heart, Hepatitis, Tumor, Chess, Letter, Segment, Sick, Sonar, and Splice (Murphy and Aha, 1994). These datasets are from a variety of problem domains and have different combinations of nominal and numerical attribute values. Some have missing values and noisy data.

Test Results

Tables 2 and 3 display the test results. A few observations can be made from the evidence presented in these tables:

1. No one particular entropy measure stands out above all the others over the collection of datasets used in the study.

2. Some datasets are insensitive to the choice of splitting methods (e.g. *Page Blocks, Mushroom, Chess, Letter, Segment, Sick*) while other datasets are sensitive to the choice of splitting methods, with some being extremely sensitive (e.g. *Splice, Audiology, Glass, Soybean, Heart*). An interesting observation is that by using a relaxed rule for determining which attributes are considered, EffCAMI_1 gives a performance that was significantly better than EffCAMI_0 for the *Splice* dataset.

3. In some cases, such as with the *Splice* dataset, the within-family differences were greater than the differences in the best performances of both families. In other cases, such as with the *Heart* dataset, one family outperformed the other.

4. EffCAMI_1 and CAIR may perform better than GR on datasets where all non-class attributes are nominal; while GR and EffCAMI_1 performances are approximately the same when all non-class attributes are either continuous or integer (i.e. {C}, {I}, {C,I}).

5. If the dataset only consists of continuous non-class attributes and the splitting method only does binary discretization then there is no difference between CAMI and CAIR, because for each attribute CAIR = CAMI/2, and so for each node of the DT the choice of attribute based on CAIR would be the same as that based on CAMI.

6. The CAMI family of splitting methods performed impressively compared to the more popular CE family.

Given these observations it seems appropriate to suggest that splitting methods from the CAMI family should be included in data mining toolsets. Although the question might be raised as to which splitting method should be selected by the data miner, it should be noted that it is never known beforehand which splitting method will lead to the best DT for the given dataset. Many modern data mining tools provide multiple options for splitting methods. For example SAS Enterprise Miner offers the data miner the option of selecting either ChiSquared, Entropy (i.e. Gain Ratio), or Gini. Also modern data mining tools typically provides the data miner with multiple parameters (e.g. depth of DT, pre-pruning & post-pruning rules) with multiple options for each. Gersten et. al. (2000) notes that with regard to setting parameter values, there is "no practicable approach to select … the most promising combinations early in the process" and as such "it is necessary to experiment with different combinations" in order to be able to reliably pick the best DT. The process of DT induction in an industrial setting thus involves experimentation with different combinations of parameter settings and in some cases with different training and validation datasets in order to be able to select the most appropriate decision tree. Therefore, given that data miners already experiment with different splitting methods, it would be worthwhile to include methods from a family that performs impressively against the currently most popular method.

Further, if the given dataset is relatively small it might be possible to apply several methods and then use the one that gives the best performance with regard to criteria such as accuracy, stability, and simplicity. If the given dataset is relatively large, it might be very costly to explore the performance of several splitting methods on the entire dataset. One approach in this case is to take a stratified sample from the given dataset, apply the different splitting methods to this sample, and then apply the splitting method that gave the best performance to the entire dataset. It should be noted that such an approach is also used in industrial applications, and as such some commercial data mining tools provide convenient facilities for sampling of the dataset.

Conclusion

In this paper we have conducted a computational exploration of the performance of two families of entropy-based splitting methods, Conditional Entropy and Class-Attribute Mutual Information (CAMI), using C4.5-like algorithms on test datasets from a variety of problem domains. These results suggested that while some datasets are insensitive to the choice of splitting methods, that some of the CAMI family methods may be more appropriate than GR for datasets where all non-class attributes are nominal; that if the only type of discretization on continuous attributes is binarization then some of the CAMI methods perform as well as GainRatio for datasets where all the non-class attributes are either integer or continuous; and that the new EffCAMI_1 method and the older CAIR method performed very well, particularly when compared to the popular Gain Ratio method. Finally, we suggested strategies to pursue when faced with choosing the appropriate measure to use when performing data mining on real datasets, large and small.

Table 2. Classification Accuracy of CE and CAMI Families

Dataset		CE Family		CAMI Family					
ID	**Name**	**Gain Ratio**	**Cond. Entropy**	**CAMI**	**CAIR**	**EffCAMI_0**	**EffCAMI_1**	**Best-Worst**	**Winning Family**
1	IRIS	**95.33**	94.67	94.67	94.67	94.67	94.67	0.66	CE
2	Breast Cancer	72.49	72.49	72.49	72.49	**74.50**	**74.50**	2.01	CAMI
3	Credit Approval	85.94	84.20	85.36	**86.96**	84.64	84.64	2.76	CAMI
4	Car	92.48	93.52	93.52	**93.52**	92.48	92.48	1.04	CAMI
5	Abalone	20.19	20.79	20.76	20.71	20.40	**21.88**	1.69	CAMI
6	Wave	**77.02**	**76.74**	**76.76**	**76.76**	75.66	75.24	1.78	**TIE**
7	Glass	65.89	67.76	67.76	67.76	**71.96**	69.63	6.07	CAMI
8	Soybean	92.09	87.56	89.02	89.02	92.09	**92.68**	5.12	CAMI
9	Page Blocks	96.95	96.99	96.99	96.99	96.97	96.78	0.21	**TIE**
10	Mushroom	100.00	100.00	100.00	100.00	100.00	100.00	0.00	**TIE**
11	Wine	94.94	**95.51**	**95.51**	**95.51**	94.94	94.94	0.57	**TIE**
12	Yeast	54.78	53.03	53.03	53.03	**55.39**	54.72	2.36	CAMI
13	Zoo	92.08	**94.06**	**94.06**	**94.06**	92.08	92.08	1.98	**TIE**
14	Pima	**74.09**	72.14	72.14	72.14	71.48	71.61	2.61	CE
15	Nursery	97.11	97.10	97.10	**98.13**	97.11	97.11	1.03	CAMI
16	Audiology	77.88	67.26	77.43	77.43	77.88	**79.65**	12.39	CAMI
17	Heart	**77.78**	72.59	72.59	72.59	73.33	73.33	5.19	CE
18	Hepatitis	79.35	78.06	**80.65**	**80.65**	80.00	80.00	2.59	CAMI
19	Tumor	40.71	43.01	40.41	40.41	42.18	**44.25**	3.84	CAMI
20	Chess	99.53	99.44	99.44	99.44	99.47	99.47	0.09	**TIE**
21	Letter	87.76	87.96	87.96	87.96	87.68	87.67	0.29	**TIE**
22	Segment	97.14	97.10	97.10	97.10	96.93	96.97	0.21	**TIE**
23	Sick	98.65	98.52	98.97	98.91	98.67	98.67	0.45	**TIE**
24	Sonar	74.04	73.08	73.08	73.08	**75.97**	**75.96**	2.89	CAMI
25	Splice	**93.98**	51.88	51.88	**93.51**	51.88	**93.67**	42.10	**TIE**

****TIE**** indicates that the difference in the Classification Accuracy is 0.50% or less

Table 3. Selected Pairwise Comparisons of Splitting Methods

Dataset		Comparison							
ID	Non-Class Attribute Data Type(s)	GR vs CE	GR vs EffCAMI_0	GR vs EffCAMI_1	GR vs CAIR	CAIR vs EffCAMI_0	CAIR vs EffCAMI_1	CE vs CAIR	CAMI vs CAIR
1	C	GR	GR	GR	GR	TIE	TIE	TIE	TIE
6	C	TIE	GR	GR	TIE	CAIR	CAIR	TIE	TIE
7	C	CE	EffCAMI_0	EffCAMI_1	CAIR	EffCAMI_0	EffCAMI_1	TIE	TIE
11	C	CE	TIE	TIE	CAIR	CAIR	CAIR	TIE	TIE
12	C	GR	EffCAMI_0	TIE	GR	EffCAMI_0	EffCAMI_1	TIE	TIE
24	C	GR	EffCAMI_0	EffCAMI_1	GR	EffCAMI_0	EffCAMI_1	TIE	TIE
9	C, I	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
14	C, I	GR	GR	GR	GR	CAIR	CAIR	TIE	TIE
17	C, I	GR	GR	GR	GR	EffCAMI_0	EffCAMI_1	TIE	TIE
22	C, I	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
23	C, I	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
2	I	TIE	EffCAMI_0	EffCAMI_1	TIE	EffCAMI_0	EffCAMI_1	TIE	TIE
21	I	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
4	N	CE	TIE	TIE	CAIR	CAIR	CAIR	TIE	TIE
8	N	GR	TIE	EffCAMI_1	GR	EffCAMI_0	EffCAMI_1	CAIR	TIE
10	N	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
15	N	TIE	TIE	TIE	CAIR	CAIR	CAIR	CAIR	CAIR
16	N	GR	TIE	EffCAMI_1	TIE	TIE	EffCAMI_1	CAIR	TIE
19	N	CE	EffCAMI_0	EffCAMI_1	TIE	EffCAMI_0	EffCAMI_1	CE	TIE
20	N	TIE	TIE	TIE	TIE	TIE	TIE	TIE	TIE
25	N	GR	GR	TIE	TIE	CAIR	TIE	CAIR	CAIR
5	N, C	CE	TIE	EffCAMI_1	CAIR	TIE	EffCAMI_1	TIE	TIE
3	N, C, I	GR	GR	GR	CAIR	CAIR	CAIR	CAIR	CAIR
18	N, C, I	GR	EffCAMI_0	EffCAMI_1	CAIR	CAIR	CAIR	CAIR	TIE
13	N, I	CE	TIE	TIE	CAIR	CAIR	CAIR	TIE	TIE

C: Continuous **I**: Integer **N**: Nominal

References

Bradley, P., Fayyad, Usama M., Mangasarian O. L., "Mathematical Programming for Data Mining: Formulations and Challenges," *INFORMS Journal on Computing*, vol. 11, no. 3, 1999, pp. 217-238.

Bryson, K-M. "On Two Families of Entropy-based Splitting Methods", Working Paper, Department of Information Systems, Virginia Commonwealth University, USA, 2000.

Cheeseman,P., Stutz, J., "Bayesian Classification (AutoClass): Theory and Results," in Gregory Piatetsky-Shapiro Usama Fayyad, Padhraic Smyth, ed., *Advances in Knowledge Discovery and Data Mining*, Menlo Park: AAAI Press, MIT Press, 1996, pp. 153-180.

Ching, J., Wong, A., Chan, K., "Class-Dependent Discretization for Inductive Learning from Continuous and Mixed-Mode Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, 1995, pp. 631-641.

Gersten, W., Wirth, D. and Arndt, D. (2000) "Predictive Modeling in Automative Direct Marketing: Tools, Experiences and Open Issues", *Proceedings of 2000 International Conference on Knowledge Discovery and Data Mining (KDD-2000)*, Boston, MA, pp.398-406.

Murphy, P., Aha, D. W., UCI Repository of Machine Learning Databases. 1994, University of California, Department of Information and Computer Science:

Piatetsky-Shapiro, G. "The Data-Mining Industry Coming of Age," *IEEE Intelligent Systems*, vol. 14, no. 6, 1999, pp. 32-34.

Quinlan, J. "Induction of Decision Trees," *Machine Learning*, vol. 1, 1986, pp. 81-106.

Quinlan, J. *C4.5 Programs for Machine Learning*, San Mateo: Morgan Kaufmann, 1993.

Safavian, S., Landgrebe, D., "A Survey of Decision Tree Classifier Methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, 1991, pp. 660-674.

Wu, X., Urpani, D., "Induction by Attribute Elimination," *IEEE Transactions on Knowledge and Data Engineering*, vol. 11, no. 5, 1999, pp. 805-812.