A Kolmogorov-Smirnoff Metric for Decision Tree Induction

Paul E. Utgoff
Jeffery A. Clouse

Technical Report 96-3
January 10, 1996

Department of Computer Science
University of Massachusetts
Amherst, MA 01003

Telephone: (413) 545-4843
Net: utgoff@cs.umass.edu, clouse@cs.umass.edu

Contents

Abstract

In 1977, Friedman demonstrated that Kolmogorov-Smirnoff distance could be employed effectively as a test selection metric for decision tree induction. We revisit this metric and modify it to handle multiple classes within a single tree, and to be sensitive to missing data values. Empirical results for a large sample of learning tasks, comparing this metric to the gain ratio metric, show a highly significant reduction in tree size and expected number of tests for classification, without a significant change in classification accuracy.

1 Introduction

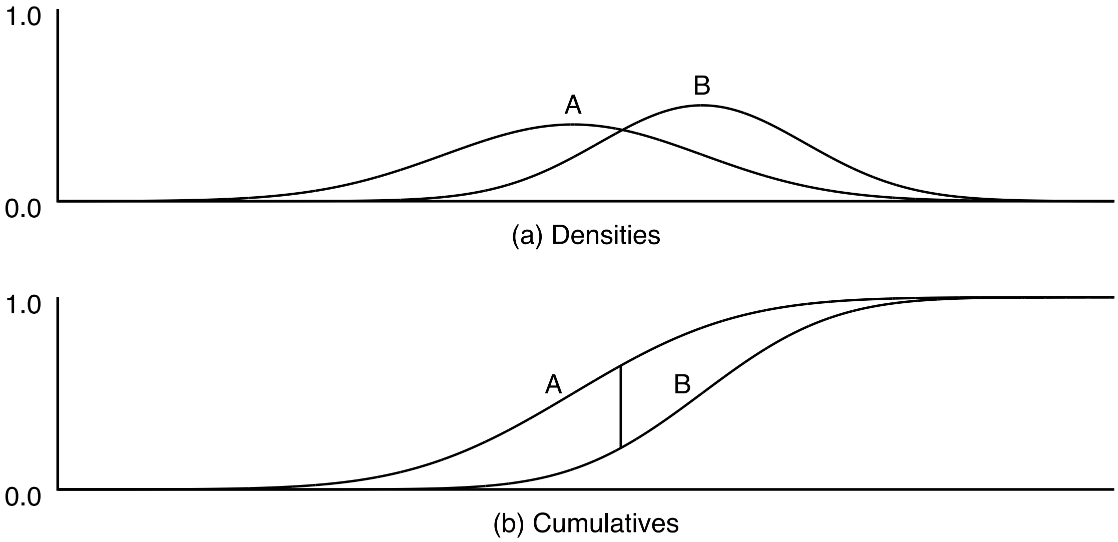
Top-down induction of decision trees is driven by greedy selection of a partition of the training instances that maximizes a heuristic function of that partition. The heuristic function is often called the *test selection metric*, but it is also known as the *splitting criterion*, the *attribute selection metric*, and the *partition merit function*. A good test selection metric should have a higher value for a better partition, but whether one partition is indeed better than another is the subject of some debate. Nevertheless, it is generally accepted that a purer partition is probably better than one that is less pure. For example, a partition for which each block contains training instances of a single class is desirable because the test that produces that partition is probably a good predictor for other instances whose labels are currently unknown.

A variety of test selection metrics have been proposed, including *gain* and *gain ratio* (Quinlan, 1993), *gini* (Breiman, Friedman, Olshen & Stone, 1984), *ORT* (Fayyad & Irani, 1992), and *distance* (de Mántaras, 1991). We revisit and extend a metric proposed by Friedman (1977) based on Kolmogorov-Smirnoff distance. The metric is attractive for theoretical reasons, raising the question of how well it works in practice. We extend the definition of the metric to handle multiclass problems in a single tree, and we modify the manner in which missing values are handled during tree construction.

2 Kolmogorov-Smirnoff Distance

The definition of Kolmogorov-Smirnoff distance, hereafter called KS distance, appears in Friedman (1977), Rounds (1980), and Gordon and Olshen (1978), and is restated here for convenience. Consider the two-class case, and assume for the moment that we need to find a single cut-point α that partitions a continuous variable x's values into a two-block partition. One block contains those values of x for which $x < \alpha$, and the other block contains the remaining values. Assume further that we have available the class-conditional probability density functions $f_A(x)$ and $f_B(x)$ for classes A and B respectively. Finally, assume that the misclassification costs for classes A and B are identical, and that the prior probabilities of drawing an instance from a particular class are identical. Then a Bayes-optimal cutpoint α is a value that minimizes the probability that the test (decision rule) will produce an incorrect classification.

Consider the cumulative distribution functions $F_A(x)$ and $F_B(x)$ that correspond to $f_A(x)$ and $f_B(x)$ respectively. As illustrated in Figure 1, an optimal cutpoint α is one that maximizes $|F_A(\alpha) - F_B(\alpha)|$, and this maximum value is the KS distance for that variable. Thus, when one computes KS distance for a variable, one also locates a Bayes-optimal cutpoint



(a) Densities

(b) Cumulatives

Figure 1. Class-Conditional Probability Densities and Cumulatives for Two Classes

for that variable, under the above assumptions. In the same way that one picks an optimal test given a particular variable, one picks an optimal test across all the variables by picking the test with the maximum KS distance across all the variables. Under the stated assumptions, there is no better split available for a single-test classifier. The variable, and the test based upon it that produce the maximum KS distance, indicate the best test to place at the decision node during tree construction.

One does not normally possess analytical versions of the densities f_A and f_B, nor of the cumulatives F_A and F_B, but they can be approximated by inspecting the training instances and estimating them from the observed frequencies. Because the calculation of KS distance involves only the cumulative distributions, we need only consider approximations $\hat{F}_A$ and $\hat{F}_B$. This can be done in the standard way for a candidate cutpoint, for example as practiced in C4.5, by counting the number of instances in each class that fall into each of the blocks of the candidate partition. For example, as shown in Figure 2, one can count the number of cases of each class in each block, and convert these frequencies to the class-conditional cumulative distributions for a candidate test, in this case giving a distance of 0.3. For convenience, we refer to one block of the paritition as L and the other as R.

3 A Modified Kolmogorov-Smirnoff Metric

We have described KS distance as a test selection metric in its simplest form. Now we modify its definition and application.

Class	L	R
A	60	40
B	30	70

(a)

Class	L	R
A	.6	.4
B	.3	.7

(b)

Figure 2. Frequencies converted to Class-Conditional Cumulatives

3.1 Discrete Variables

It is a simple matter to apply the KS distance computation to any discrete binary partition. One can see that in the continuous case described above, the cumulative distributions for a proposed test are approximated by counting how many instances from each class fall into each block of the partition. The continuous case was handled by synthesizing a boolean variable $x < \alpha$. So, we already know how to handle a binary variable. For any binary test, one can produce immediately the approximated cumulatives $\hat{F}_A$ and $\hat{F}_B$.

For a discrete variable with more than two possible values, one can enumerate the binary tests, for example those of the form $x = v_i$, and compute the distance for each one. The test that maximizes the KS distance is the optimal test for that variable under the stated assumptions. The process is entirely analogous to the continuous case, for which enumeration of the possible tests was also done.

Hereafter, we no longer distinguish whether a variable is continuous or discrete, nor do we concern ourselves with the particular method of enumerating the possible tests based on that variable. Generically, one picks the best single test, across all the variables, by enumerating the possible tests and selecting the one with the greatest KS distance. That distance is the KS distance for the partition that would be formed by that test.

3.2 Uneven Class Distributions

Bayes decision theory defines an optimal classification rule in terms of the posterior class-conditional probabilties. The use of KS distance described above assumes that the prior probabilities of membership in each class are identical. The class-conditional density functions have equal area, and this is true for this case even when multiplied by the prior probabilities, as is done to convert to posterior probabilities. However, when the prior probabilities are different, so too are the areas of the posterior probability distributions. In principle, this does not change the definition of KS distance, because one can still use the same method to compute the distance between the corresponding cumulative distributions (Gordon & Olshen, 1978). This breaks down however because the densities would be scaled, which would mean that they were no longer densities, thereby making the cumulatives incomparable.

This raises the question of whether to take into account the prior probabilities when computing KS distance for the purpose of test selection in decision tree induction. Consider the uneven class distribution shown in Figure 3(a). One can see that the probability of an instance being in class A is $P(A) = \frac{10}{1010}$ and that the probability for class B is $P(B) = \frac{1000}{1010}$.

Class	L	R
A	9	1
B	100	900

(a)

Class	L	R
A	.9	.1
B	.1	.9

(b)

Figure 3. Uneven Class Distribution

Given the large number of instances from class B, one might argue that the test is not informative because one would want to classify an unlabeled instance as belonging to class B regardless of the outcome of the test. However, the test is indeed informative. If the result of the test is outcome L, then $P(A|L)$ is $\frac{9}{109}$, whereas if the result of the test is outcome R, then $P(A|R)$ is $\frac{1}{901}$, which is very much lower. Knowing the result of the test improves our probability estimate for each of the classes, even though it may not affect choosing one for the purpose of classification.

We use the approximated class-conditional cumulative distributions to compute KS distance *without* making use of the prior probability of each class. Friedman assumes equal prior probabilities, and accordingly samples evenly from each class. Our approach is nearly the same, but we ignore the prior probabilities as irrelevant for selecting a test. This is a design choice based on the goal of partitioning the instances based solely on the result of a test. Note in Figure 3(b) that the corresponding class-conditional cumulatives for these observed frequencies indicate that the test itself separates two quite different regions of certainty regarding the predicted class.

A second reason that we choose not to take the class distribution into account at every decision node is that the class distribution is an artifact of the partition that was formed by the test that was selected at the node's parent and its ancestors. Except at the root node, the distribution is contrived. One should not repeatedly extract information from the class distributions.

3.3 Multiclass Tree Induction

How can one extend the KS distance metric to handle more than two classes? One can, as Friedman suggests, build one tree for each class. An alternative approach is to build a single tree, and extend the metric to handle multiple classes.

A single optimal cutpoint α can be found by searching for the variable value that produces the largest distance between two adjacent-valued cumulative distributions, as illustrated in Figure 4. However, since our goal is to produce a good partition, not a single-stage classifier, we know that additional recursive partitioning will separate instances further as necessary. So, for the purpose of picking a single good test, one does not need to distinguish classes that are predominantly on the same side of the candidate cutpoint. In the figure, one does not need to distinguish between classes A and B, so one can instead group them into a superclass, suggesting a form of twoing (Breiman, Friedman, Olshen & Stone, 1984).

To find a good cutpoint for a multiclass split, we do the following. For each candidate

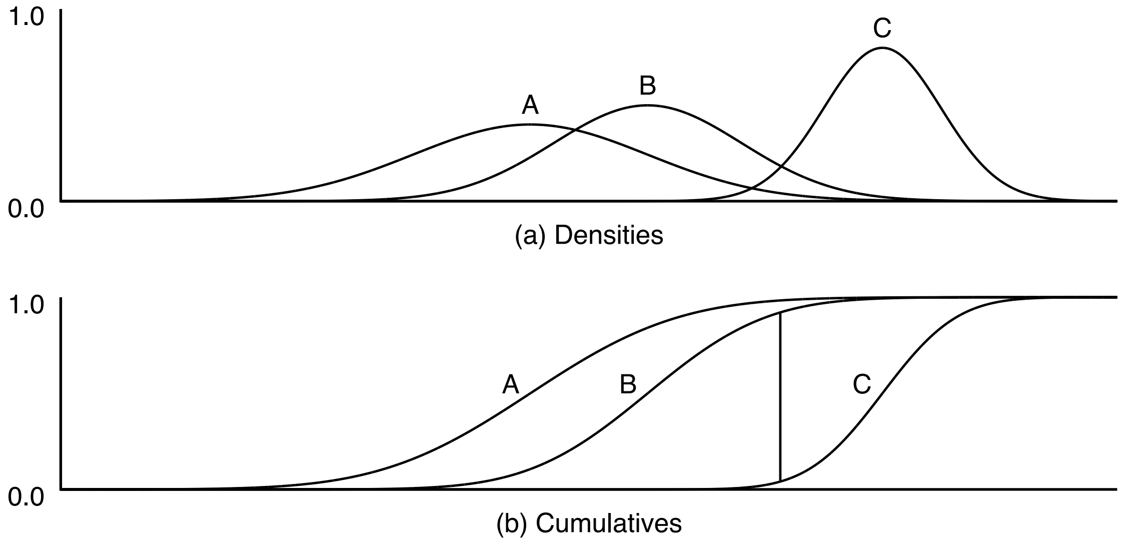


Figure 4. Class-Conditional Probability Densities and Cumulatives for Three Classes

value of x to be considered as a cutpoint α, sort the class-conditional cumulative probabilties. Then find the two adjacent cumulative probabilities with the largest difference. Then group the frequencies for those classes with cumulative probabiliies at least as high as the higher of these two adjacent probabilities into one superclass, and group the remaining frequencies into the second superclass. Finally, compute the distance between the two superclass cumulatives as before in the two-class case. As before, the α that maximizes the difference between the superclass cumulative distributions is deemed best, and that distance is the KS distance for that variable. We cannot currently say in what sense, if any, this cutpoint is optimal, but it is faithful in spirit to the KS distance formulation.

Figure 5 illustrates the process. Part (a) shows the class-conditional cumulatives from which the grouping of classes is decided. Part (b) shows the corresponding frequencies before grouping, and part (c) shows the frequencies after grouping. Finally, part (d) shows the class-conditional cumulatives for the superclasses, giving a KS distance of 0.69 for this candidate partition.

3.4 Missing Values

One needs to allow for the not infrequent case in which the value of a variable is missing from the training instance. One can simply omit such an instance when estimating probability from frequency, but then the class-conditional probability estimates become less well founded. One would rather incorporate this degradation in an explicit manner. To this end, we include the count of the instance in the denominator, but not the numerator, when es-

Class	L	R
A	.9	.1
B	.8	.2
C	.2	.8

(a)

Class	L	R
A	90	10
B	8	2
C	40	160

(b)

Class	L	R
A∨B	98	12
C	40	160

(c)

Class	L	R
A∨B	.89	.11
C	.20	.80

(d)

Figure 5. Multi-Class KS Distance

Class	L	R	?
A	6	3	1
B	3	5	2

(a)

Class	L	R
A	.6	.3
B	.3	.5

(b)

Figure 6. KS Distance with Missing Values

timating probability from frequency. This amounts to treating a missing value as a distinct value for the purpose of computing KS distance for the variable.

The process is illustrated in Figure 6. Part (a) shows the frequencies, including those with missing values. The corresponding cumulatives are shown in part (b). Notice that the KS distance computed for outcome L is 0.3 yet the KS distance computed for outcome R is 0.2. Without missing values, these distances are always identical, but they can differ when using this computation that accounts for missing values. We compute the distance for both blocks and sum the two numbers. The overall effect is that missing values reduce the KS distance. In the extreme, when the value is missing in every instance, the KS distance is diminished all the way to zero.

4 Empirical Comparison with Gain Ratio

In order to evaluate the modified Kolmogorov-Smirnoff metric, which we shall call KS2 here, we compared it to the standard gain ratio metric. To control for the influence of other factors that arise in inducing decision trees, such as pruning method and binary versus multi-way splits, we employed the ITI algorithm (Utgoff, 1994; Utgoff, 1995) in its batch mode, varying only the test selection metric. Consequently, the performance differences that we measured are due solely to the differences in these two metrics.

We selected a sample of 27 tasks from the UCI repository (Murphy & Aha, 1994). For each task we ran a ten-fold cross validation experiment. For each fold, a tree was built while using the gain ratio test selection metric, and a tree was built while using the KS2 test selection metric. In each fold, 90% of the instances were used for training, and the remaining 10% were used for testing. For each of the cross validation runs we measured classification accuracy on the held-out testing instances, the number of nodes in the induced tree, and

Table 1. Results of Cross-Validation Runs

	Clasification Accuracy		Number of Nodes		Expected Number of Tests	
Tasks	ks2	gr	ks2	gr	ks2	gr
balance-scale	74.28±3.80	76.51±3.87	310.20±4.13	276.60±7.76	6.25±0.10	7.02±0.16
breast	94.14±2.28	94.00±3.49	58.40±3.66	73.60±5.50	3.48±0.19	7.10±0.34
bupa	67.43±4.51	59.72±8.24	144.80±3.94	176.00±13.70	5.96±0.07	21.58±2.02
chess-551x39	89.29±2.38	92.32±1.89	173.40±15.83	141.20±12.38	6.86±0.25	9.45±0.45
cleveland	73.55±10.29	71.94±9.38	86.20±4.34	102.60±9.23	4.77±0.10	8.35±1.08
crx	79.43±3.03	80.28±7.73	155.80±6.55	184.20±13.93	5.65±0.19	11.12±1.27
glass-no-id	66.36±9.63	66.36±12.16	87.60±5.89	86.80±6.21	7.21±0.49	9.21±0.77
hepatitis	81.88±6.88	75.63±9.52	27.80±3.29	38.00±9.39	2.81±0.22	5.24±1.24
hypothyroid	98.71±0.67	98.64±0.63	66.80±6.56	66.80±7.45	2.76±0.24	4.01±0.38
image	96.60±1.07	96.91±1.18	129.20±20.75	135.20±27.14	6.16±1.18	6.88±1.12
ionosphere	85.83±7.79	93.89±3.88	41.40±3.24	45.00±3.27	4.07±0.13	10.67±0.50
iris	95.00±4.93	94.38±5.47	16.60±2.46	17.40±2.27	2.53±0.13	2.89±0.20
led7	67.62±11.83	67.62±12.66	98.20±3.68	91.00±3.89	6.21±0.11	5.81±0.12
lung-cancer	45.00±25.82	42.50±23.72	20.40±3.27	21.00±3.77	3.69±0.35	4.79±1.35
lympho	78.67±10.33	77.33±10.52	52.40±4.12	60.00±5.60	5.96±0.17	8.08±0.42
mplex-11	100.00±0.00	100.00±0.00	183.40±43.10	183.40±43.10	6.09±0.33	6.09±0.33
mushroom	100.00±0.00	100.00±0.00	23.00±0.00	24.80±0.63	2.66±0.04	5.03±0.13
nettalk	83.47±1.09	83.81±1.16	1762.60±42.27	1679.60±24.62	23.65±0.62	26.72±0.24
pima	71.56±3.94	69.09±3.17	245.60±7.24	303.00±19.55	6.54±0.08	17.35±3.06
post-op	54.00±10.75	56.00±10.75	70.40±6.19	70.60±7.71	6.23±0.29	7.78±0.42
promoter	71.82±10.00	77.27±12.31	22.20±2.35	23.60±3.78	3.07±0.16	3.94±0.65
soybean	83.91±3.24	91.45±3.01	114.60±4.09	134.80±12.35	6.82±0.04	8.51±0.27
splice	92.94±1.31	91.72±0.90	263.80±7.79	283.20±7.33	6.34±0.08	7.32±0.10
votes	93.64±4.77	93.64±4.39	43.60±4.84	45.20±4.02	2.81±0.12	2.88±0.13
vowel	81.60±3.31	79.20±4.87	295.00±8.43	308.00±16.20	11.12±0.42	13.75±0.73
waveform	70.65±9.67	64.84±13.21	67.20±5.45	92.20±19.30	5.30±0.28	13.03±3.10
wine	92.22±10.21	94.44±5.24	14.60±2.27	14.60±2.27	3.21±0.39	3.08±0.40

Table 2. ANOVA results

Measurement	KS2	Gain Ratio	Significance
Accuracy	81.10%	81.09%	**not**
Number of Nodes	169.45	173.27	$p \ll 0.01$
Expected Tests	5.86	8.80	$p \ll 0.01$

the expected number of tests needed to classify an instance. The averages and standard deviations of these measurements for all 27 tasks appear above in Table 1.

One can observe that neither KS2 nor gain ratio dominate with respect to the accuracy of the induced trees. For example, both metrics produce similar accuracy measurements for the BREAST task. Sometimes KS2 produces higher accuracy, such as for the HEPATITIS task, and sometimes gain ratio produces higher accuracy, such as for the SOYBEAN task.

Regarding the total number of nodes in the induced trees, there is a marked imbalance toward cases in which KS2 produces smaller trees than gain ratio. Indeed, in only 6 of 27 cases does gain ratio produce a smaller tree.

Similarly, with respect to the expected number of tests in the induced trees, those produced with the KS2 metric require noticeably fewer than those produced with the gain ratio metric. In only two cases does a tree produced with the gain ratio metric have fewer expected tests (LED7 and WINE). The number of expected tests is an important measure because it

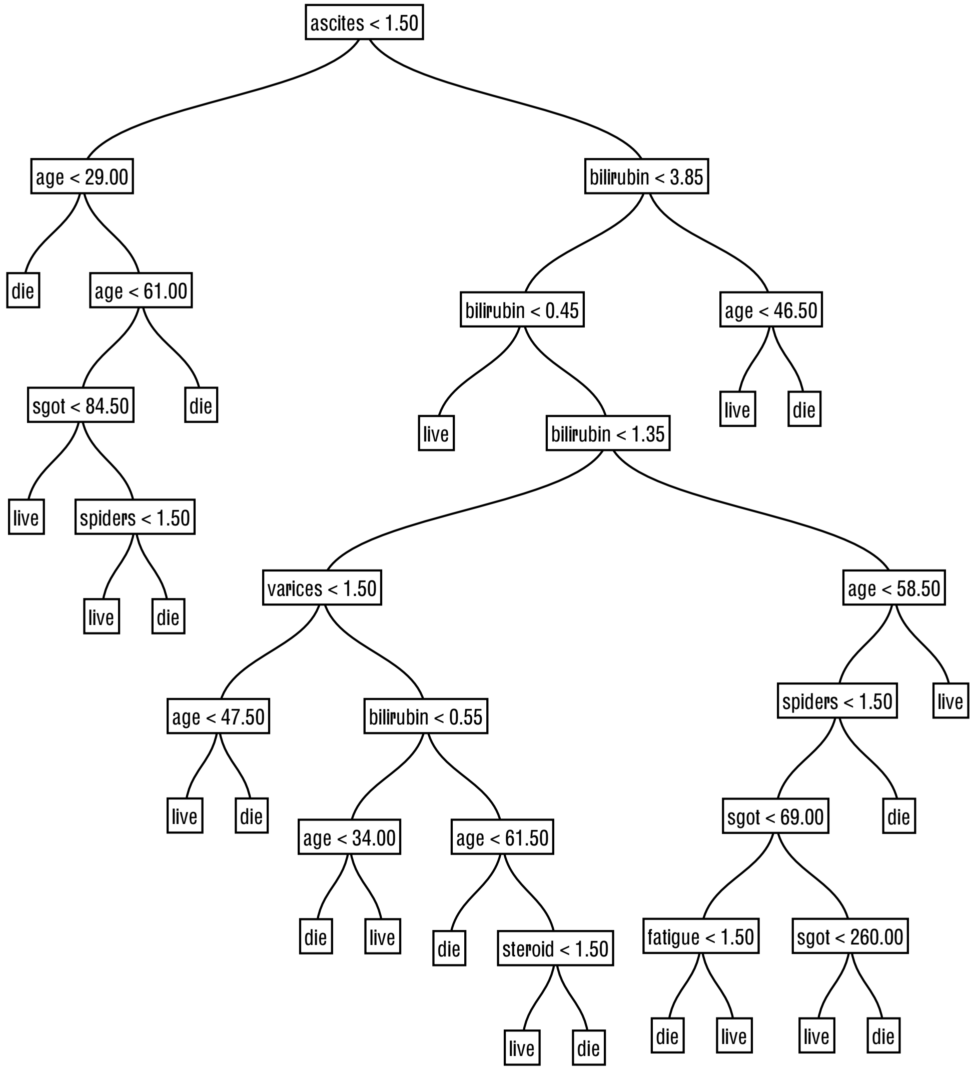


Figure 7. Hepatitis Tree via Gain Ratio (branch left on True)

provides an indication of the number of tests that one must perform on average to determine a classification.

Statistical analysis supports these observations. After accumulating the measurements for each task and metric, we performed a two-way analysis of variance on each of the three measures of interest. The average accuracy measured for the KS2 metric is 81.10% while for the gain ratio metric it is 81.09%. According to the ANOVA, one cannot reject the null hypothesis that these two means are different, which indicates that the difference in accuracy is not statistically significant. However, the same analysis applied to the remaining two measures indicates that the differences in the means of each are highly significant ($p \ll 0.01$). One concludes that the trees built with the KS2 metric contain significantly fewer total nodes, and require significantly fewer tests (on average) to classify an instance. These results are summarized above in Table 2.

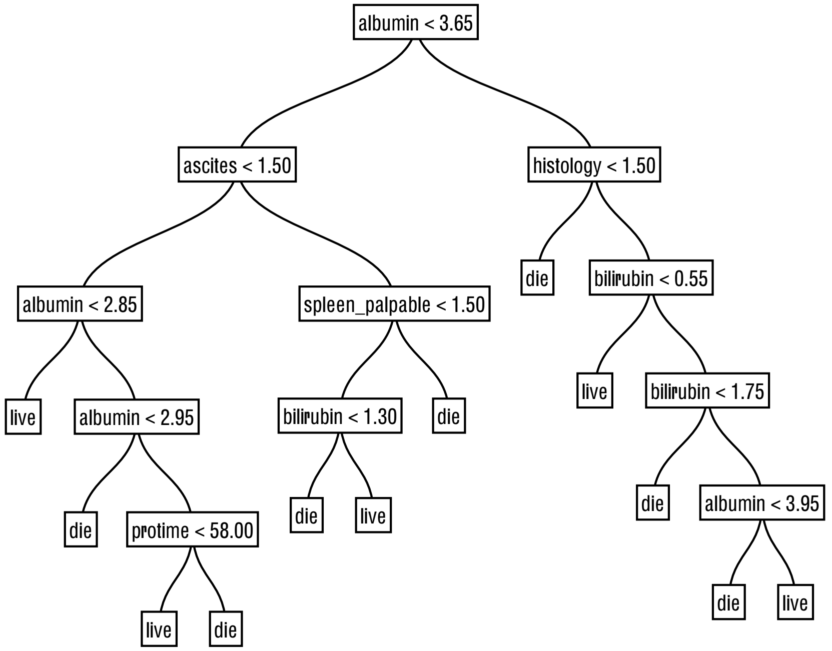


Figure 8. Hepatitis Tree via KS2 (branch left on True)

5 Tree Quality

To see whether we could discern any difference in the quality of tree produced by the two different metrics, we selected the two trees built for one of the folds of the hepatitis task. This selection was based on the desire to look at trees that were small in size. The tree built when using the gain ratio metric (the GR tree) is shown in Figure 7, and the corresponding tree built when using the KS2 metric (the KS2 tree) is shown in Figure 8. The GR tree is larger and deeper than the KS2 tree. These trees were shown to a medical doctor, who picked the KS2 tree as matching her sense of liver failure diagnosis much better than the GR tree. For example, the KS2 tree tests principally albumin, bilirubin, and protime, whereas the GR tree tests minor criteria such as varices, sgot, spiders, and fatigue. Although one would like to do a more thorough comparison of the trees built by these two metrics, it is clear from the significantly smaller size and expected number of tests that the kinds of trees that are built are quite different.

6 Summary

We have revisited Friedman's approach to using Kolmogorov-Smirnoff distance as a test selection metric. We modified the metric to handle multiple classes within a single tree, and to handle missing values in a manner that diminishes the value of the metric as the number of missing values increases. We noted that the method can be applied directly to discrete variables, and we argued that the metric should not take into account the class distributions at a node.

We showed empirically, for a large sample of learning tasks, that the modified metric KS2 produces trees that are significantly smaller, and require significantly fewer tests to classify instances, than trees produced with the gain ratio metric. This comes about with no loss in classification accuracy.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. IRI-9222766. We thank Doina Precup and Gunnar Blix for helpful comments. We thank Michele VanderHeyden for commenting on the hepatitis trees. The lymphography data was provided to UCI by M. Zwitter and M. Soklic of the University Medical Center at the Institute of Oncology in Ljubljana.

References

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees.* Belmont, CA: Wadsworth International Group.

de Mántaras, R. L. (1991). A distance-based attribute selection measure for decision tree induction. *Machine Learning, 6*, 81-92.

Fayyad, U. M., & Irani, K. B. (1992). The attribute selection problem in decision tree generation. *Proceedings of the Tenth National Conference on Artificial Intelligence* (pp. 104-110). San Jose, CA: MIT Press.

Friedman, J. H. (1977). A recursive partitioning decision rule for nonparametric classification. *IEEE Transactions on Computers, C-26*, 404-408.

Gordon, L., & Olshen, R. A. (1978). Asymptotically efficient, computationally feasible solutions to the classification problem. *Annals of Statistics, 6*, 515-533.

Murphy, P. M., & Aha, D. W. (1994). *UCI repository of machine learning databases*, Irvine, CA: University of California, Department of Information and Computer Science.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning.* Morgan Kaufmann.

Rounds, E. M. (1980). A combined nonparametric approach to feature selection and binary decision tree design. *Pattern Recognition, 12*, 313-317.

Utgoff, P. E. (1994). An improved algorithm for incremental induction of decision trees. *Machine Learning: Proceedings of the Eleventh International Conference* (pp. 318-325). New Brunswick, NJ: Morgan Kaufmann.

Utgoff, P. E. (1995). *Decision tree induction based on efficient tree restructuring*, (Technical Report 95-18), Amherst, MA: University of Massachusetts, Department of Computer Science.