Matching between partition incidence matrices and decision tree classifier criteria

Boris Mirkin[1,2] and Azizbek Bekniyazov[1]

[1]School of Data Analysis and Artificial Intelligence, National Research University "Higher School of Economics", Moscow, RF
[2]School of Mathematical and Computer Sciences, Birkbeck University of London UK
bmirkin@hse.ru

Abstract. This paper proposes a family of scoring criteria for building classification trees. Our family is generated over rescaling weights of target categories according to a model matching the input and target partitions. We prove that popular Gini and Chi-squared criteria belong to the family. We experimentally show, using both real-world and synthetic datasets, that use of match scoring criteria can lead to superior results over those obtained using Entropy and other popular criteria.

Keywords: Classifier, classification tree, scoring criteria, partition match model.

1 Introduction

Classifier decision tree model is popular because of its logical and visual clarity. An important part of the feature-splitting process for tree building is the criterion for choosing both a feature to split and a cut-off over the feature. Three popular criteria: Entropy, Gini index and Pearson's Chi-squared, constitute the core of most popular decision tree producing programs, C4.5 [10], CART [2] and CHAID [4], respectively. For the time being, several significant improvements to them have been proposed including methodologies for building oblique and multivariate cut-off criteria [7]; investigation of statistical nature of the criteria with probability modeling [5]; models and methods for building optimal cut-off trees rather than just single splits [3]. Yet, however, nothing has been done with respect to investigation of theoretical fundament for the splitting criteria. Indeed, the criteria have been picked up from fragments of different frameworks, and no general framework to generate criteria – the more so to compare them – has been developed so far.

This paper makes a step in the direction of overcoming the gap. We present a simple linear model relating the target partition with the predictor partition which is being built. This model involves a set of parameters for scaling frequencies of target categories. We derive a formulaic expression for the contribution of the predictor partition to the summary variance of the target categories, which depends on the scaling parameters and, in this way, generates a family of association indices which can be used as

classification decision tree criteria. We prove that the Gini and Chi-square indices belong to this family. We pick up several other criteria from our family and experimentally verify that some of them can be superior at some real-world and synthetic datasets.

2 A model for matching partition incidence matrices

Consider a finite set I of N objects $i \in I$ and two partitions, $R = \{R_1, R_2, \dots, R_K\}$ and $S = \{S_1, S_2, \dots, S_V\}$, over I. Assign 0/1 incidence matrices, $Z = (z_{ik})$ of dimension $N \times K$ and $X = (x_{iv})$ of dimension $N \times V$, to these partitions, R and S, respectively. Elements of these matrices are defined so that $z_{ik} = 1$ if $i \in R_k$, and $z_{ik} = 0$ if $i \notin R_k$; and, similarly, $x_{iv} = 1$ if $i \in S_v$, and $x_{iv} = 0$ if $i \notin S_v$. In what follows these two play different roles. We assume that partition S, as well as matrix X, correspond to a target categorical feature over I, while partition R, as well as its matrix Z, correspond to a cluster partition over I which is to be devised to predict, or approximate, feature X from some other data whose composition is not important at these definitions.
Then the matrix X can be considered a quantified form of the target feature obtained by enveloping its categories into "dummies", the binary columns of matrix X. In analogy with (bi)linear cluster models described in [8,9], one can relate these partitions with a model like

$$y_{iv} = \sum_{k=1}^{K} z_{ik} c_{kv} + e_{iv} \quad (1)$$

where y_{iv} are elements of matrix X standardized according to equations

$$y_{iv} = \frac{x_{iv} - a_v}{b_v}$$

in which a_v is a shift parameter, and b_v a rescaling parameter for the 0/1–dummy representing the v-th part of partition S, and e_{iv} are residuals to be minimized over unknown c_{kv} values. Assume that the scale shift a_v is the average value in v-th column of matrix X, which is, obviously, the proportion of part R_v in I, further denoted as p_v. One may refer to model (1) as R-S-match model. By minimizing the sum of squares of residuals e_{iv} in (1), we arrive at expression $F(S, c) = \sum_{k=1}^{K} |S_k| < c_k, c_k >$ for that part of the data scatter T, which is explained by the association between R and S according to model in (1), to be maximized. This can be further reformulated as

$$F(R, S) = N \sum_{k=1}^{K} \sum_{v=1}^{V} \frac{(p_{kv} - p_k p_v)^2}{p_k b_v^2} \quad (2)$$

Where p_k, p_v and p_{kv} are, respectively, the proportion of objects in S_k, in R_v, and $S_k \cap R_v$. Further on we consider its relative part, $f(R, S) = \frac{F(R,S)}{N}$, which is a family generated by various values of the rescaling factor b_v.
Obviously, at $b_v = p_v^{1/2}$, the value (2) is equal to the Chi-squared Pearson association index used in the popular CHAID decision tree package [4]. Moreover, it is not difficult to prove that another popular association measure, the Gini index used in CART [2], is also a member of the family, at $b_v = 1$.
We consider a few more specifications for the b_v parameter in (2); four of them are presented in Table 1 to supplement its rows 1 and 2 corresponding to already mentioned Pearson and Gini indices.

Table 1. Values of the scaling b_v parameter in the match association criterion in (2).

No	b_v
1	1
2	$p_v^{0.5}$
3	$[p_v(1-p_v)]^{0.5}$
4	$-\log(p_v)$
5	$p_v^{0.5}\log(p_v)$
6	$-p_v\log(p_v)$

3 Setting of Experiments

The goal of experimental computations is to check whether any match association index from the list in Table 1 can be better at building classifier decision trees than any of the three popular criteria mentioned in Section 1: the Entropy, the Gini index (number 1 in Table 1), and the Chi-squared (number 2 in Table 1). Besides the set of six classification criteria in Table 1 plus Entropy, the experiment comprises:

1. Datasets at which decision trees are to be built.
2. An algorithm for building classification trees.
3. Criteria for assessment of the results.

Let us consider them in turn.

3.1 Datasets.

We use both real-world data and synthetic data. Real-world datasets are obtained from the UCI Irvine repository [1]. The list of them, together with their characteristics, is presented in Table 2. Due to space limitations, only a subset of the experimental results is reported in this paper; complete results for all the datasets are available at GitHub site https://github.com/buxazich/Datagenerator

We generate synthetic data sets by using data generator proposed in [6], see also [11]. It produces an $N \times V$ data matrix X of Gaussian clusters in the following steps:

A. We specify the number of objects, N, the number of features, V, and the number of clusters, K. We also define the minimum cluster size, n, which may be needed to operate with stochastic data models. The product $K * n$ is to be less than N: there remain $N - K * n$ objects to randomly distribute among the K clusters to be generated. In this way, the number of elements N_k is specified at each of the clusters to be generated.
B. For each cluster k, $k = 1, 2, \dots, K$, its center c_k is generated as a V-dimensional vector of components generated from a uniform distribution over interval

$(\alpha - 1,\ 1 - \alpha)$ where α is a user-defined real, $0 < \alpha < 1$, to characterize the intermix of clusters generated. The larger the squeeze parameter α, the nearer to each other are the centers and the more interspersed are elements of different clusters.

Table 2. Datasets from UCI Irvine repository used in our experiments

	Dataset	*N*	*V*	*K*
1	Breast Cancer (Prognostic)	194	32	2
2	Balance Scale	625	4	3
3	Car Evaluation	1728	6	4
4	Connectionist Bench Sonar	208	60	2
5	Contraceptive Method Choice	1473	9	3
6	Hayes Roth	132	4	3
7	Hepatitis	80	19	2
8	Ionosphere	351	34	2
9	Glass	214	9	6
10	Congressional Voting Records	232	16	2
11	Spambase	4601	57	2
12	Mushroom	8214	22	7

C. To generate k-th cluster, we generate a vector of standard deviations $g_k = (g_{kv})$ randomly from interval [0.05, 0.10], $k = 1, 2, \ldots, K$. Then we generate N_k V-dimensional vectors whose v-th components are taken from a Gaussian distribution $N(0, g_{kv})$ with 0 expectation and standard deviation g_{kv}, and add each of them to the center c_k. In this way, we obtain a matrix X_k of dimension $N_k \times V$ of elements of cluster k.
D. We combine the matrices X_k together into a final $N \times V$ data matrix X, so that the hidden partition of row indices is known to the user.

In the follow-up, all the synthetic datasets have $N = 2000$ objects; the minimum cluster size is defined as 50. There are two feature space dimensions $V = 6$ and 15, three cluster numbers $K = 4, 8$, and 15, and two squeezing parameter values $\alpha = 0.5$ (clusters well separated) and $\alpha = 0.85$ (clusters are well interspersed). Therefore, there are 12=2*3*2 parameter settings.

3.2 Computation algorithm

For each parameter combination, we generate 50 datasets. At each of the datasets, we run a binary tree classifier algorithm at the number of splits not more than $t = 3$ or $t = 4$, with the size of a cluster to be split not less than 10. The cluster to be split is chosen according to its contribution to the data scatter, computed according to formula $w_k = |S_k|\langle c_k, c_k \rangle$. The split is performed on feature v and its value r_v to maximize the scoring function, either entropy or one of functions 1-6 in Table 1. After this is done over all 50 data tables generated, we report the average value and standard deviation for each of the scoring functions. At each of the real-world datasets, we split it randomly into two subsets, the training one, 75%, of the objects, and the testing one, 25% of the objects. We build the decision tree over the training set, and then we compare that with the target partition on the testing set. Then we average them over 50 independent runs.

3.3 Assessment of results.

To quantitatively assess the performance of classification and clustering algorithms, a set of evaluation metrics is commonly employed, each characterizing different aspects of model behavior and predictive reliability.
We follow in this to the approaches developed in machine learning. When the classification task involves just two classes, positive and negative, the outcomes of binary classification are conventionally counted as true positives or true negatives if they are correct, and false negatives or false positives, if they are not. By denoting the respective numbers by TP, TN, FN, and FP, the popular evaluation metrics are Accuracy, Recall, Precision and the harmonic average of the latter two, F1. Target partitions usually have several classes (categories). In such a case, the metrics above apply at each class taken as the positive while all the rest are considered as negative, after which the results are averaged. We also apply a genuine partition-to-partition similarity metric, the popular Adjusted Rand Index (ARI) [12].

4 Experimental results

Overall, the goal of our computations was to see whether our match association indexes are relevant at all. In this section, we present evidence for the positive answer to the question, Moreover, we will see that the offhand indexes 3 through 6 in Table 1 can be superior to over the popular ones at some specific situations.

4.1 Results at real-world data.

In what follows, we report average scoring results over 50 runs of the algorithm; standard deviations are rather small, reaching not more than 10-15% of the averages, and, thus, are omitted from the tables. First, we notice that all b_v variants achieve identical performance and consistently outperform the Entropy baseline across all five assessment scores (Table 3).

An interesting case emerges at the real-world dataset Car Evaluation (Table 4). At decision trees of depth 3, there is always one winner overall. However, when we move to a better association by increasing the tree depth to 4, the picture changes considerably (see Table 4). There is no single winner anymore. Version 6. $b_v = -p_v \log(p_v)$ wins

Table 3. Average assessment scores for the Spambase dataset at tree depth =3.

Spambase							
Metric	**Entropy**	b_v					
		1	**2**	**3**	**4**	**5**	**6**
Accuracy	0.869	**0.882**	**0.882**	**0.882**	**0.882**	**0.882**	**0.882**
Precision	0.876	**0.883**	**0.883**	**0.883**	**0.883**	**0.883**	**0.883**
Recall	0.869	**0.882**	**0.882**	**0.882**	**0.882**	**0.882**	**0.882**
F1	0.867	**0.881**	**0.881**	**0.881**	**0.881**	**0.881**	**0.881**
ARI	0.543	**0.582**	**0.582**	**0.582**	**0.582**	**0.582**	**0.582**

over Accuracy and Recall. Entropy wins over Precision, while versions 3. $b_v = [p_v(1-p_v)]^{0.5}$, 4. $b_v = \log(p_v)$, and 5. $b_v = p_v^{0.5}\log(p_v)$, win over both F1 and ARI.
The reported results show that indeed, any of our offhand scoring functions can be superior to popular measures at appropriate datasets.

Table 4. Average assessment scores for the Car Evaluation dataset at tree depth =4.

Car Evaluation							
Metric	**Entropy**	b_v					
		1	**2**	**3**	**4**	**5**	**6**
Accuracy	0.834	**0.844**	0.842	**0.844**	**0.844**	**0.844**	**0.844**
Precision	**0.852**	0.845	0.843	0.846	0.846	0.846	0.845
Recall	0.834	**0.844**	0.842	**0.844**	**0.844**	**0.844**	**0.844**
F1	0.831	0.837	0.834	**0.838**	**0.838**	**0.838**	0.837
ARI	0.651	0.682	0.667	**0.684**	**0.684**	**0.684**	0.679

4.2 Results at synthetic datasets.

The following Tables 5 – 6 present patterns of match association scores at relatively modest number of features, 6, when moving from 4-part target partitions (Table 5) to 15-part target partition (Table 6).

Table 5 corresponds to the case at which clusters are well separated, and the number of features is greater than the number of target clusters, so that the decision tree approximates the target partition very closely, so that all the scoring functions reach

0.98 or 0.99, that are very close to the maximum value 1. The winning match association functions here are versions 3. $b_v = [p_v(1-p_v)]^{0.5}$ and 5. $b_v = p_v^{0.5}\log(p_v)$ from Table 1, already mentioned with respect to Table 4.

Table 5. Average assessment scores for the synthetic dataset at 6 features, 4 well separated clusters, and tree depth = 3.

***V*, *K*, α, depth = (6, 4, 0.5, 3)**							
Metric	**Entropy**	$\boldsymbol{b_v}$					
		1	**2**	**3**	**4**	**5**	**6**
Accuracy	0.992	0.993	0.993	**0.994**	**0.994**	**0.994**	0.993
Precision	0.992	0.993	0.993	**0.994**	**0.994**	**0.994**	**0.994**
Recall	0.992	0.993	0.993	**0.994**	**0.994**	**0.994**	0.993
F1	0.992	0.993	0.993	**0.994**	**0.994**	**0.994**	0.993
ARI	0.979	0.982	0.982	**0.983**	**0.983**	**0.983**	0.982

Table 6 relates to increasing the number of target clusters to a large value, $K = 15$, while leaving the number of features the same, 6. This leads to dramatic fall in precision, to 0.3 values, thus decreasing F1 value as well. There is no single winner here: Entropy is the best over Accuracy, Recall and ARI whereas scoring criterion 4. $b_v = \log(p_v)$ is the best over Precision and F1.

Table 6. Average assessment scores for the synthetic dataset at 6 features, 15 well separated clusters, and tree depth =3.

***V*, *K*, α, depth = (6, 15, 0.5, 3)**							
Metric	**Entropy**	$\boldsymbol{b_v}$					
		1	**2**	**3**	**4**	**5**	**6**
Accuracy	**0.512**	0.457	0.470	0.486	0.508	0.460	0.469
Precision	0.308	0.290	0.327	0.302	**0.349**	0.291	0.325
Recall	**0.512**	0.457	0.470	0.486	0.508	0.460	0.469
F1	0.371	0.332	0.358	0.355	**0.391**	0.333	0.358
ARI	**0.567**	0.432	0.408	0.490	0.490	0.443	0.404

These examples show that there should be some regularities over synthetic data caused by an interplay of three main forces: the dataset dimension *V*, the number of target clusters *K*, and the squeezing parameter α.

5 Conclusion

This paper proposes a family of scoring criteria to supplement the currently popular criteria of Entropy, Gini index and Chi-squared for building classification trees. Our family is generated by a least-squares model to match the input and target partitions

represented by their incidence matrices. We prove that both Gini and Chi-squared criteria belong to the match association family. We propose a few offhand match scoring criteria belonging to our family and we experimentally show that some of them can be superior to those popular ones in some situations such as, for example, when the number of target classes is larger than the number of features.

However, the results of the computational experiments depend on the characteristics of the datasets used. Therefore, the generalizability of the observed improvements may be limited and further validation on a broader range of experiments is needed.

Among directions for further development, we can indicate the issue of finding conditions at which this or that representative of our match association criteria family is superior to other criteria for classification tree building. Another direction would be developing algorithms for enhancing classification tree techniques by using representatives of our match association family. This can involve such directions as random forest classifiers or boosting.

The authors are grateful to the anonymous reviewers for helpful comments. Support from the Basic Research Program of HSE University is gratefully acknowledged.

References

1. Asuncion, A., and Newman, D. (2007, November). UCI Machine Learning Repository.
2. Breiman, L., Friedman, J. H., Olshen, V. and Stone, C. J. Classification and Regression Trees. Wadsworth, Belmont, 1984.
3. Demirović, E., Lukina, A., Hebrard, E., Chan, J., Bailey, J., Leckie, C., ... and Stuckey, P. J. (2022). Murtree: Optimal decision trees via dynamic programming and search. *Journal of Machine Learning Research*, *23*(26), 1-47.
4. Kass, G.V. (1980). An exploratory technique for investigating large quantities of categorical data. Applied Statistics, 29, 119-127.
5. Klusowski, J. M., and Tian, P. M. (2024). Large scale prediction with decision trees. *Journal of the American Statistical Association*, *119*(545), 525-537.
6. Kovaleva, E. V., and Mirkin, B. G. (2015). Bisecting K-means and 1D projection divisive clustering: A unified framework and experimental comparison. *Journal of Classification*, *32*(3), 414-442. https://doi.org/10.1007/s00357-015-9186-y
7. Loh, W. Y. (2014). Fifty years of classification and regression trees. *International Statistical Review*, *82*(3), 329-348. https://doi.org/10.1111/insr.12016
8. Mirkin, B. G. (1990). A sequential fitting procedure for linear data analysis models. *Journal of Classification*, *7*(2), 167-195. https://doi.org/10.1007/bf01908715
9. Mirkin B. (2019) Core data analysis, Springer.
10. Quinlan, J. R. (2014). *C4.5: Programs for Machine Learning*. Elsevier.
11. Rykov, A., De Amorim, R. C., Makarenkov, V., and Mirkin, B. (2024). Inertia-based indices to determine the number of clusters in k-means: an experimental evaluation. *IEEE Access*, *12*, 11761-11773.
12. Steinley, D. (2004). Properties of the Hubert-Arable adjusted Rand index. *Psychological Methods*, *9*(3), 386. https://doi.org/10.1037/1082-989X.9.3.386
13. Sut, N., and Simsek, O. (2011). Comparison of regression tree data mining methods for prediction of mortality in head injury. *Expert Systems with Applications*, *38*(12), 15534-15539. https://doi.org/10.1016/j.eswa.2011.06.006