

Classification and Regression Tree Methods

(In *Encyclopedia of Statistics in Quality and Reliability*,
Ruggeri, Kenett and Faltin (eds.), 315–323, Wiley, 2008)

Wei-Yin Loh

Department of Statistics

University of Wisconsin

Madison, WI 53706

loh@stat.wisc.edu

Phone: 1 608 262 2598; Fax: 1 608 262 0032

Keywords: cross-validation, discriminant, linear model, prediction accuracy, recursive partitioning, selection bias, unbiased

Abstract

A classification or regression tree is a prediction model that can be represented as a decision tree. This article discusses the C4.5, CART, CRUISE, GUIDE, and QUEST methods in terms of their algorithms, features, properties, and performance.

1 INTRODUCTION

Classification and regression are two important problems in statistics. Each deals with the prediction of a response variable y given the values of a vector of predictor variables \mathbf{x} . Let \mathcal{X} denote the domain of \mathbf{x} and \mathcal{Y} the domain of y . If y is a continuous or discrete variable taking real values (e.g., the weight of a car or the number of accidents), the problem is called regression. Otherwise, if \mathcal{Y} is a finite set of unordered values (e.g., the type of car or its country of origin), the problem is called classification. In mathematical terms, the problem is to find a function $d(\mathbf{x})$ that maps each point in \mathcal{X} to a point in \mathcal{Y} . The construction of $d(\mathbf{x})$ requires the existence of a *training sample* of n observations $\mathcal{L} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$. In computer science, the subject is known as *supervised learning*. The criterion for choosing $d(\mathbf{x})$ is usually mean squared prediction error $E\{d(\mathbf{x}) - E(y|\mathbf{x})\}^2$ for regression, where $E(y|\mathbf{x})$ is the expected value of y at \mathbf{x} , and expected misclassification cost for classification.

If \mathcal{Y} contains J distinct values, the classification solution (or *classifier*), may be written as a partition of \mathcal{X} into J disjoint pieces $A_j = \{\mathbf{x} : d(\mathbf{x}) = j\}$ such that $\mathcal{X} = \cup_{j=1}^J A_j$. A *classification tree* is a special form of classifier where each A_j is itself a union of sets, with the sets being obtained by recursively partitioning the \mathbf{x} -space. This permits the classifier to be represented as a decision tree. A *regression tree* is similarly a tree-structured solution in which a constant or a relatively simple regression model is fitted to the data in each partition.

Table 1: Variables in the car dataset

Name	Definition	Values (#unique values in parentheses)
Region	Manufacturer region	Asia, Europe, or U.S. (3)
Make	Make of car	Acura, Audi, etc. (38)
Type	Type of car	Car, minivan, pickup, sports car, sport utility vehicle, or wagon (6)
Drive	Drive type	Front, rear, or four-wheel drive (3)
Rprice	Suggested retail price	U.S. dollars (410)
Dcost	Dealer cost	U.S. dollars (425)
Engnsz	Size of engine	liters (43)
Cylin	Number of cylinders	-1 for the rotary-engine Mazda RX-8 (8)
Hp	Horsepower	hp (110)
City	City miles per gallon	miles (29)
Hwy	Highway miles per gallon	miles (32)
Weight	Weight of car	pounds (347)
Whlbase	Length of wheel base	inches (40)
Length	Length of car	inches (66)
Width	Width of car	inches (18)

To illustrate, consider a dataset on new cars for the 2004 model year from the *Journal of Statistics Education Data Archive* (www.amstat.org/publications/jse/jse_data_archive.html). After filling in missing values and correcting errors, we obtain a dataset with complete information on 15 variables for 428 vehicles. Table 1 gives the definitions of the variables and Figure 1 shows two classification trees for predicting the **Region** variable, i.e., whether the vehicle manufacturer is from Asia, Europe, or the U.S. (there are 158 Asian, 120 European, and 150 U.S. vehicles in the dataset). Each intermediate node has a condition associated with it. If an observation satisfies the condition, it goes down the left branch; otherwise it goes down the right branch. Each leaf node is labeled with a predicted class, with cross-hatched, light gray, and dark gray denoting Asia, Europe, and the U.S., respectively. Beneath each node is a fraction, giving the number of misclassified samples divided by the number of samples. It is easily seen from the trees that **Dcost** is an important predictor for European cars: 125 of 159 cars with **Dcost** greater than \$30,045 are European. For cars with **Dcost** less than or equal to \$30,045, those with 3.5-liter or larger engines are six times as likely to be American, while those with smaller engines are predominantly Asian.

A classification or regression tree algorithm has three major tasks: (i) how to partition the data at each step, (ii) when to stop partitioning, and (iii) how to predict the value of y for each \mathbf{x} in a partition? There are many approaches to the first task. For ease of interpretation, a large majority of algorithms employ *univariate* splits of the form $x_i \leq c$ (if x_i is non-categorical) or $x_i \in B$ (if x_i is categorical). The variable x_i and the split point c or the split set B are often found by an exhaustive search that optimizes a *node impurity criterion* such as entropy (for classification) or sum of squared residuals (for regression). There are also several ways to deal with the second task, such as stopping rules and tree pruning. The third task is the simplest: the predicted y value at a leaf node is the class that

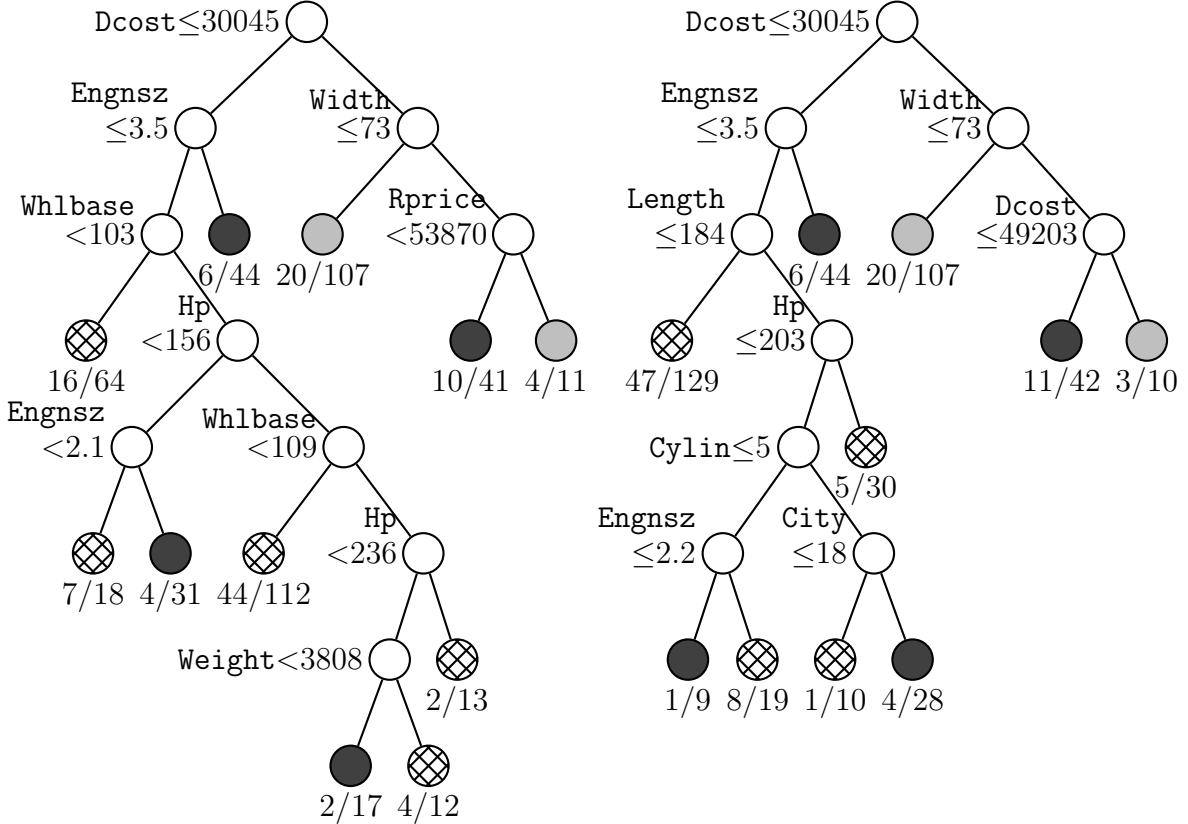


Figure 1: RPART (left) and QUEST (right) trees for `Region`. At each intermediate node, an observation goes to the left branch if and only if the condition shown at the node is satisfied. Leaf nodes that are cross-hatched, light gray, and dark gray are classified as Asia, Europe, and U.S., respectively. The fraction below a leaf node gives the number misclassified divided by the sample size. The total numbers misclassified are 119 for RPART and 106 for QUEST.

minimizes the estimated misclassification cost (for classification), or the fitted value from a model estimated at the node (for regression).

We review and compare below a few of the more established and widely available algorithms. First, we introduce some notation. Let N_j be the number of class j training samples and let $\pi(j)$ be the prior probability of class j . At each node t , let $N_j(t)$ be the number of class j training samples in t and let $p(j|t) = \pi(j)N_j(t)/N_j$ denote the estimated probability that an observation in t belongs to class j . It is easily verified that $p(j|t) \propto N_j(t)$ if $\pi(j) \propto N_j$.

2 CART and RPART

We first discuss CART [1] classification with the *Gini index* as node impurity criterion, $i(t) = 1 - \sum_{j=1}^J p^2(j|t)$. Suppose a split divides the data in t into a left node t_L and a right node t_R . Let p_L and p_R be the proportions of data in t_L and t_R , respectively. CART selects the split that maximizes the decrease in impurity $i(t) - p_L i(t_L) - p_R i(t_R)$. Instead of

employing stopping rules, CART generates a sequence of subtrees by growing a large tree and pruning it back until only the root node is left. Then it uses cross-validation to estimate the misclassification cost of each subtree and chooses the one with the lowest estimated cost. The tree on the left side of Figure 1 is produced by RPART [2], a version of CART implemented in R [3].

Like the AID and THAID [4, 5] algorithms before it, CART has some undesirable properties. First, the splitting method is biased toward variables that have more distinct values. This is because a variable with m distinct values allows $(m - 1)$ splits if it is non-categorical and $(2^{m-1} - 1)$ splits if it is categorical. The bias was first noted in [6] and is discussed in [7]. Second, the exponential number of splits for categorical variables causes serious computational difficulties when m is large and y takes more than two values (a computational trick reduces the number of splits searched to $(m - 1)$ if y takes two values). Finally, the algorithm is also biased toward variables with more missing values. This is due to the impurity function depending only on the sample proportions, not the sample sizes. During split selection, if a variable x_i has missing values, only the observations non-missing in both x_i and y are used in computing the decrease in impurity. Thus it is easier to “purify” a node by splitting on a variable with more missing values [8].

For regression, CART uses the sum of squared residuals as impurity function and builds a piecewise-constant model with each leaf node fitted by the training sample mean. Everything else remains the same as in classification. For example, Figure 2 displays a RPART tree for predicting `Hp` with variables `Region`, `Type`, `Drive`, `Engnsz`, `Cylin`, `Weight`, `Whlbase`, `Length`, and `Width`. The predicted values beneath the leaf nodes show that `Hp` tends to increase with `Cylin` and to be higher for sports cars.

Piecewise-constant regression trees tend to have lower prediction accuracy compared to other regression methods that have more smoothness. In fact, CART regression trees typically have lower accuracy than even the classical multiple linear model—see, e.g., [1, p. 227] and [9]. In the example here, the RPART model has an R^2 value of 75%, compared to 81% for the multiple linear model. One way to increase the accuracy of the piecewise-constant model is to use a larger tree, i.e., with less pruning. The correct amount of pruning is, however, usually difficult to determine. Besides, a large tree is undesirable because it is more difficult to interpret.

3 QUEST and CRUISE

QUEST [7] is a classification method. It avoids the selection bias and categorical variable computational problems of CART by first selecting the variable x_i and then selecting its split point or split set. This saves a significant amount of computation, because the algorithm does not have to search for the split points or split sets of the other variables. QUEST uses hypothesis tests to choose the split variables, namely, analysis of variance F -tests for non-categorical variables and chi-squared tests for categorical variables. The variable with the smallest significance probability is selected to split a node. If the dataset has more than two classes, they are grouped into two superclasses prior to split point or split set selection. This superclass grouping allows application of the CART two-class computational shortcut for splits on categorical variables. The QUEST tree for our dataset is shown on the right

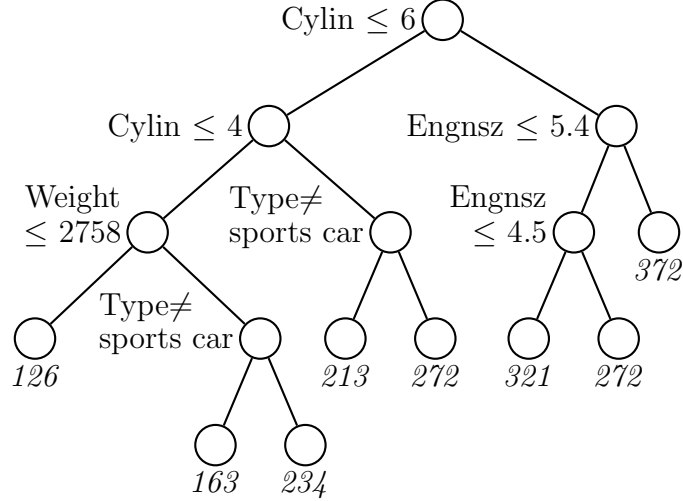


Figure 2: RPART tree for predicting Hp. The number in italics beneath each leaf node is the predicted value.

side of Figure 1. It has the same splits at the top two levels as the RPART tree.

CRUISE [8] is another classification algorithm with unbiased variable selection. It differs from QUEST in three important respects: (i) each node can be split into as many as J branches, with one class forming a majority in each branch, (ii) the significance tests include interactions between pairs of variables, and (iii) it can fit linear discriminant models in each leaf node [10]. Being able to split a node into J branches can be advantageous if J is large, because it increases the probability that at least one leaf node is assigned to each class. For example, Figure 3 shows the CRUISE tree for predicting **Region**. The vehicles in the left branch ($Dcost \leq 23256$) are mostly from Asian manufacturers and those in the right branch ($Dcost > 32775$) are mostly European.

Figure 4 shows the corresponding CRUISE tree where a linear discriminant model is fitted to the data in each leaf node. Notice how compact it is. This is due to part of the model complexity being absorbed by the discriminant models whose boundaries and data points are displayed in Figure 5. The plot for Node 3 reveals a very expensive car with $Dcost$ more than \$170,000. It is a Porsche.

4 C4.5 and J48

C4.5 [11] is also a classification tree algorithm. If the variable x_i chosen to split a node is non-categorical, the node is divided into two using a split of the usual form $x_i \leq c$. On the other hand, if the variable is categorical taking m values, the node splits into m branches, with one branch for each categorical value. As a consequence, C4.5 has no difficulty dealing with categorical variables regardless of the size of m . The algorithm chooses the splits as follows. Suppose a node t is split into subnodes t_1, t_2, \dots, t_k . Let $e(t) = -\sum_j p(j|t) \log\{p(j|t)\}$ be the entropy at t and define the *gain* of the split as $e(t) - N(t)^{-1} \sum_i e(t_i)N(t_i)$. Define the *gain ratio* as the gain divided by $N(t)^{-1} \sum_i N(t_i) \{\log N(t_i) - \log N(t)\}$. C4.5 chooses the split that yields the highest gain ratio. C4.5 also grows a large tree and then prunes it back

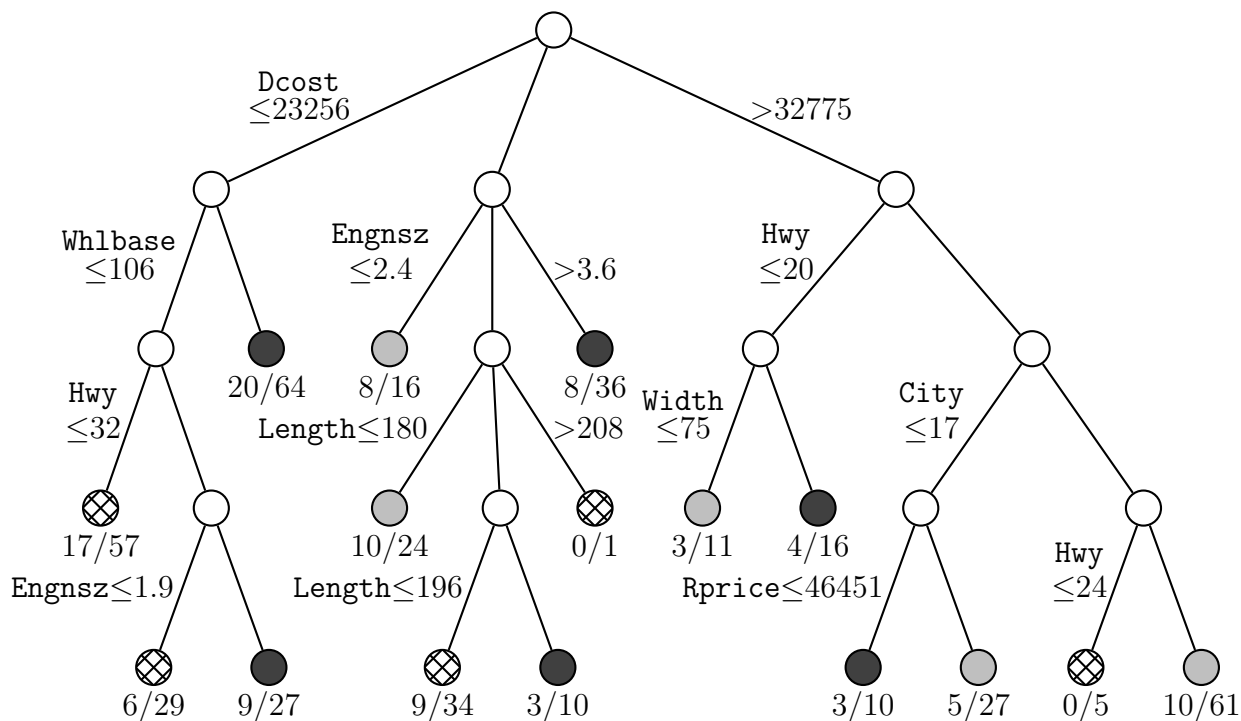


Figure 3: CRUISE tree for predicting **Region**. Cross-hatched, light gray, and dark gray leaf denotes are classified as Asia, Europe, and U.S., respectively. The fraction below a leaf node gives the number misclassified divided by the sample size. Total number misclassified is 115.

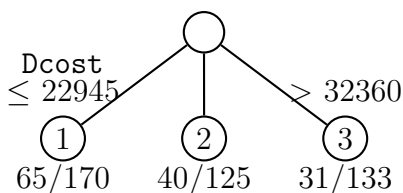


Figure 4: CRUISE tree with linear discriminant node models for predicting **Region**. The fraction below a leaf node gives the number misclassified divided by the sample size.

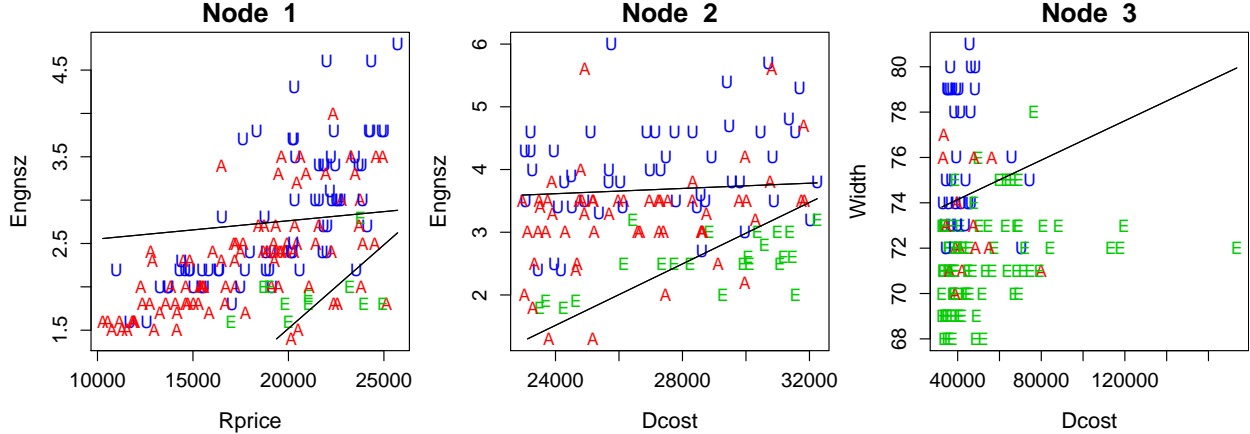


Figure 5: Data and linear discriminant boundaries for the nodes of the CRUISE tree in Figure 4. Asian, European, and U.S. cars are denoted by the symbols A, E, and U, respectively. Node 3 has only one boundary: it serves to separate the European and U.S. cars.

to a smaller size. But instead of cross-validation, it uses a conservative estimate of the error at each node to prune.

Figure 6 shows a C4.5 tree, obtained using the J48 implementation in WEKA [12], with `Drive` as the response variable and `Make` as one of the predictor variables. As anticipated, when `Make` is chosen, the node splits into many branches. This is not a problem in itself, for we can manually merge the leaf nodes associated with the same predicted class. The problem is that each node before merging has too few samples to support further splitting. Hence the tree may be too short.

The avoidance of searching for binary splits on categorical variables and of cross-validation for pruning makes C4.5 one of the fastest classification tree algorithms. Its trees, however, tend to have more leaf nodes than those of other methods [13]. Like CART, it is biased toward selecting variables that allow more splits.

5 GUIDE

GUIDE [14] constructs piecewise-constant, multiple linear, and simple polynomial tree models for least-squares, quantile, Poisson, and proportional hazards regression. Like CRUISE and QUEST, its variable selection is unbiased. This is accomplished by recursively carrying out the following steps at each node: (i) fit a model to the training data there, (ii) cross-tabulate the signs of the residuals with each predictor variable to find the one with the most significant chi-square statistic, and (iii) search for the best split on the selected variable, using the appropriate loss function. After a large tree is constructed, it is pruned with the cross-validation method of CART.

Figure 7 shows the GUIDE tree for predicting `Hp` when the best simple linear model is fitted to the data in each node (two cars with rotary engines are removed due to the `Cylin` variable being undefined for them). The sample mean value of `Hp` and the best linear predictor variable are given beneath each leaf node. The first two splits divide the tree into three main branches, with Asian, European, and American makes occupying the left,

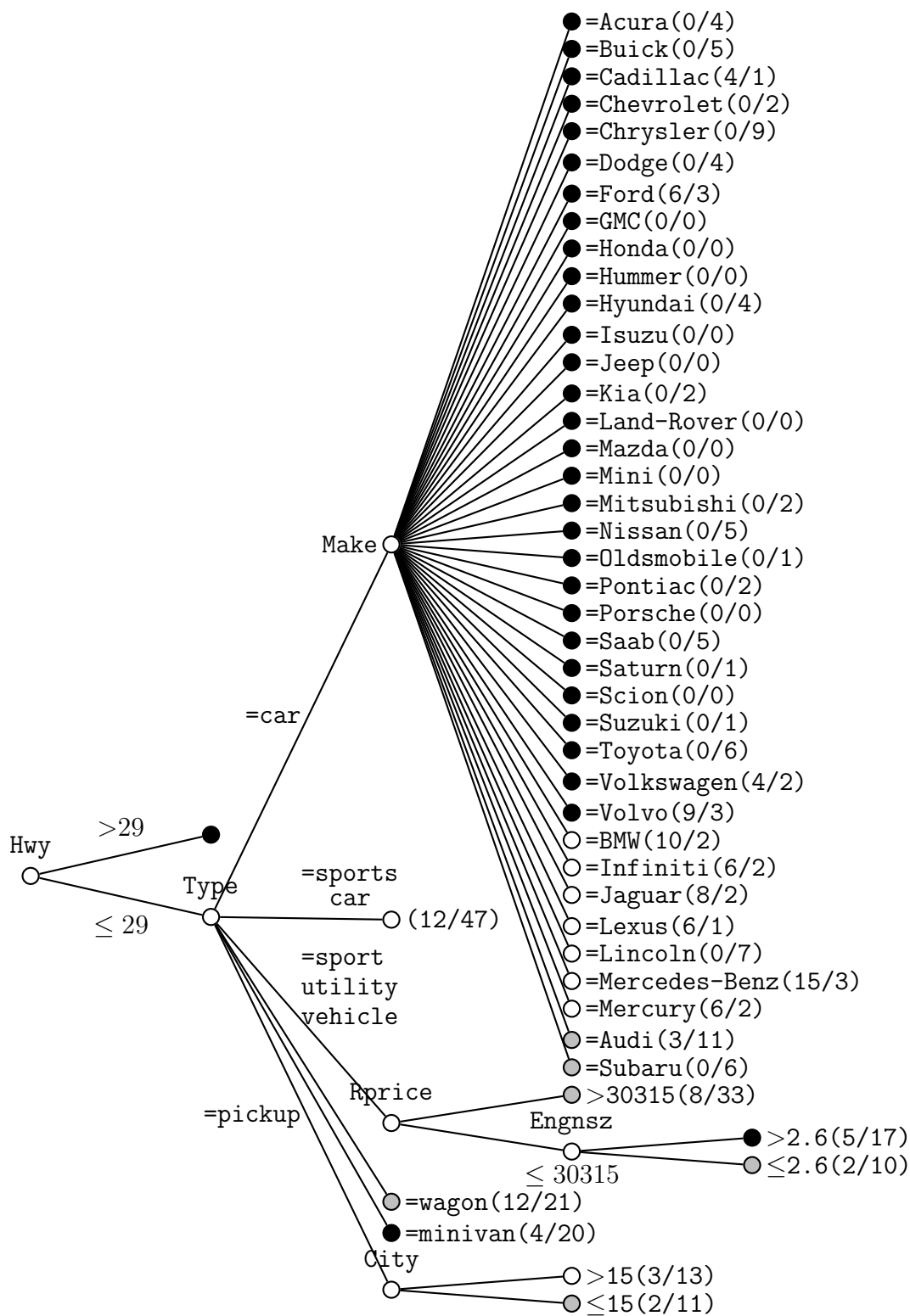


Figure 6: C4.5 (J48) tree for Drive, with black, white, and, gray-colored leaf nodes denoting front-wheel, rear-wheel, and four-wheel drive, respectively.

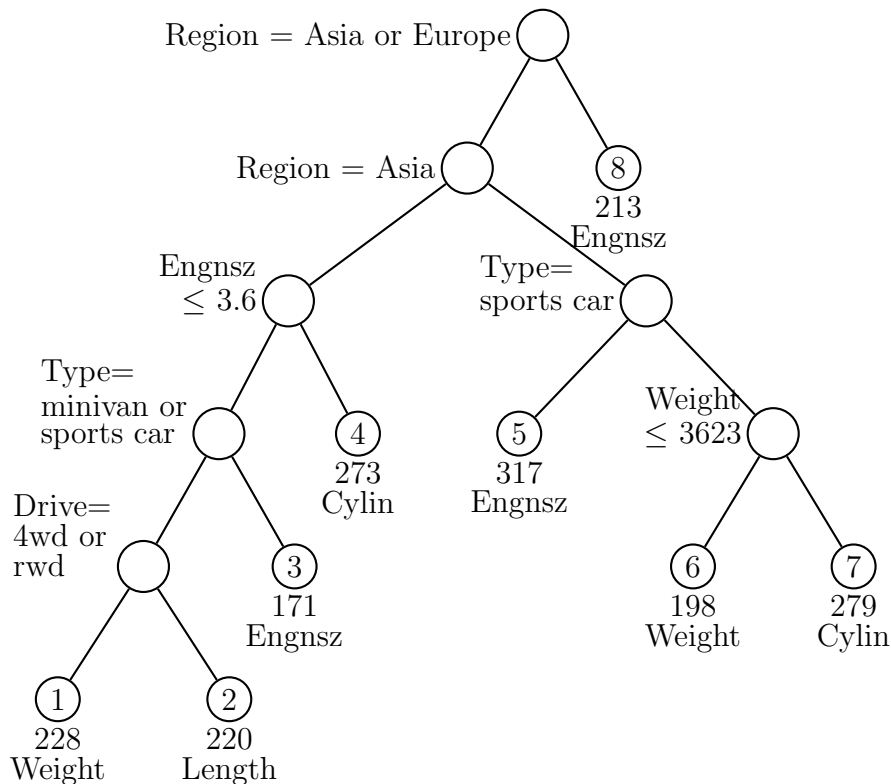


Figure 7: GUIDE piecewise simple linear model for predicting Hp. Beneath each leaf node are the sample mean of Hp and the name of the linear predictor.

middle, and right branches, respectively. The most important linear predictors are **Cylin** and **Engnsz**. A major advantage of this model, compared to a piecewise multiple linear model, is that the data and the fitted regression functions can be visualized graphically, as demonstrated in Figure 8. We see from Node 5 that European sports cars with large engines have among the highest Hp values. For American cars, Hp increases linearly with **Engnsz**, irrespective of the other variables. This GUIDE model has an R^2 value of 84%, which is higher than that of the RPART and multiple linear models discussed in Section 2.

6 CONCLUSION

All the above algorithms can deal with datasets with missing values. CART uses “surrogate splits” to pass observations with missing split values through its nodes. CRUISE follows a similar approach, but using an alternative set of splits. It is not known if one method is better than the other. C4.5 uses weights, passing an observation with a missing split value down every branch, where the weight is proportional to the number of observations non-missing that variable in the branch. GUIDE and QUEST employ nodewise plug-in estimates for the missing values.

CART, CRUISE, and QUEST accept user-specified class prior probabilities and unequal misclassification costs. Class priors are useful if the training dataset is not a random sample. The three algorithms also allow splits on linear combinations of variables. Although

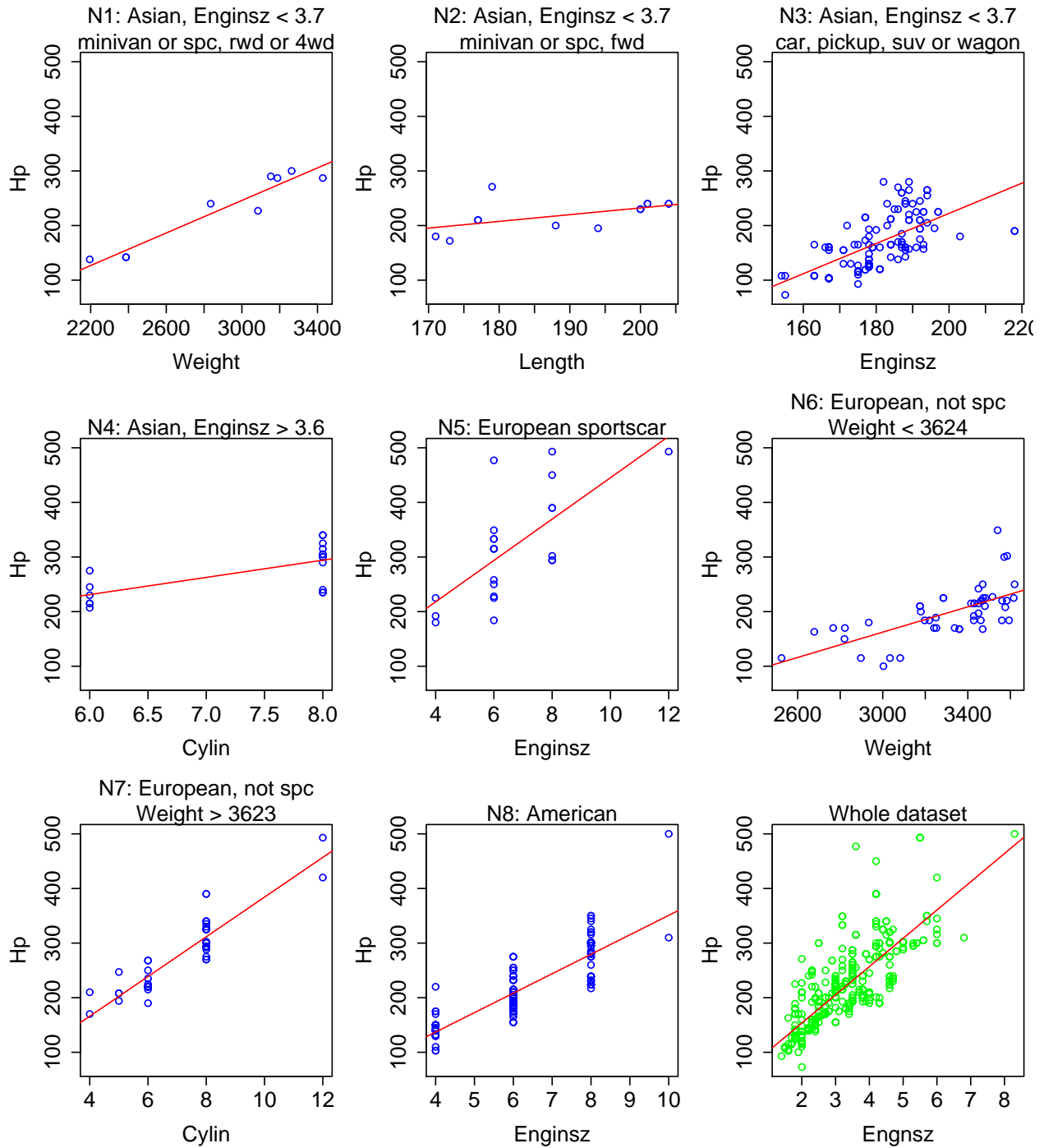


Figure 8: Data and simple linear regression models at the nodes of the GUIDE tree in Figure 7. The abbreviation `spc` stands for sports car. A plot of Hp versus Engnsz for the whole dataset is given in the bottom right corner for comparison.

such splits are hard to interpret, there is empirical evidence that they have much better prediction accuracy than trees with univariate splits [13]. C4.5 does not have these capabilities. GUIDE is also restricted to univariate splits, but empirical studies with real datasets suggest that the prediction accuracy of its piecewise multiple linear models is, on average, comparable to that of the best methods, including spline models [9].

There are many other tree algorithms that space restrictions prevent their discussion here. The interested reader is referred to CHAID [15, 16] for classification, RECPAM [17] for classification and regression, M5 [18, 19] for regression, LOTUS [20] for logistic regression, and Bayesian CART [21, 22].

In terms of statistical theory, methods based on recursive partitioning are known to be *risk consistent* under certain regularity conditions. Specifically, the expected mean squared error of piecewise-constant regression trees and the expected misclassification cost of classification trees converge to the lowest possible values as the training sample size increases [1]. Further, for piecewise linear regression trees, the predicted values converge to the unknown regression function values pointwise, at least for least-squares [23], logistic, Poisson [24], and quantile [25] regression.

7 ACKNOWLEDGMENTS

CART is a registered trademark of California Statistical Software. This article was prepared with partial support from grants from the U.S. Army Research Office and the National Science Foundation.

References

- [1] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, 1984.
- [2] T. M. Therneau and E. J. Atkinson. An introduction to recursive partitioning using the RPART routine. Technical Report 61, Mayo Clinic, Section of Statistics, 1997.
- [3] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2006. ISBN 3-900051-07-0.
- [4] J. N. Morgan and J. A. Sonquist. Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association*, 58:415–434, 1963.
- [5] A. Fielding. Binary segmentation: The automatic detector and related techniques for exploring data structure. In C. A. O’Muircheartaigh and C. Payne, editors, *The Analysis of Survey Data, Volume I, Exploring Data Structures*. Wiley, New York, 1977.
- [6] P. Doyle. The use of Automatic Interaction Detector and similar search procedures. *Operational Research Quarterly*, 24:465–467, 1973.
- [7] W.-Y. Loh and Y.-S. Shih. Split selection methods for classification trees. *Statistica Sinica*, 7:815–840, 1997.

- [8] H. Kim and W.-Y. Loh. Classification trees with unbiased multiway splits. *Journal of the American Statistical Association*, 96:589–604, 2001.
- [9] H. Kim, W.-Y. Loh, Y.-S. Shih, and P. Chaudhuri. A visualizable and interpretable regression model with good prediction power. *IIE Transactions. Special Issue on Data Mining and Web Mining*, 2007.
- [10] H. Kim and W.-Y. Loh. Classification trees with bivariate linear discriminant node models. *Journal of Computational and Graphical Statistics*, 12:512–530, 2003.
- [11] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.
- [12] I. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with JAVA Implementations*. Morgan Kaufmann, San Fransico, CA, second edition, 2005. <http://www.cs.waikato.ac.nz/ml/weka>.
- [13] T.-S. Lim, W.-Y. Loh, and Y.-S. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40:203–228, 2000.
- [14] W.-Y. Loh. Regression trees with unbiased variable selection and interaction detection. *Statistica Sinica*, 12:361–386, 2002.
- [15] G. V. Kass. Significance testing in automatic interaction detection (A.I.D.). *Applied Statistics*, 24:178–189, 1975.
- [16] G. V. Kass. An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29:119–127, 1980.
- [17] A. Ciampi, S. A. Hogg, S. McKinney, and J. Thiffault. RECPAM: A computer program for recursive partition and amalgamation for censored survival data and other situations frequently occurring in biostatistics, I: Methods and program features. *Computer Methods and Programs in Biomedicine*, 26:239–256, 1988.
- [18] J. R. Quinlan. Learning with continuous classes. In *Proceedings of the Australian Joint Conference on Artificial Intelligence*, pages 343–348, Singapore, 1992. World Scientific.
- [19] Y. Wang and I. Witten. Inducing model trees for continuous classes. In *Proceedings of the Poster Papers of the European Conference on Machine Learning*, Prague, 1997.
- [20] K.-Y. Chan and W.-Y. Loh. LOTUS: An algorithm for building accurate and comprehensible logistic regression trees. *Journal of Computational and Graphical Statistics*, 13:826–852, 2004.
- [21] H. Chipman, E. I. George, and R. E. McCulloch. Bayesian CART model search (with discussion). *Journal of the American Statistical Association*, 93:935–960, 1998.
- [22] D. G. Denison, B. K. Mallick, and A. F. M. Smith. A Bayesian CART algorithm. *Biometrika*, 85:363–377, 1998.

-
- [23] P. Chaudhuri, M.-C. Huang, W.-Y. Loh, and R. Yao. Piecewise-polynomial regression trees. *Statistica Sinica*, 4:143–167, 1994.
- [24] P. Chaudhuri, W.-D. Lo, W.-Y. Loh, and C.-C. Yang. Generalized regression trees. *Statistica Sinica*, 5:641–666, 1995.
- [25] P. Chaudhuri and W.-Y. Loh. Nonparametric estimation of conditional quantiles using quantile regression trees. *Bernoulli*, 8:561–576, 2002.