Stepwise Induction of Logistic Model Trees

Annalisa Appice, Michelangelo Ceci, Donato Malerba, and Savino Saponara

Dipartimento di Informatica, Università degli Studi di Bari
via Orabona, 4 - 70126 Bari, Italy
{appice,ceci,malerba}@di.uniba.it, savinos@email.it

Abstract. In statistics, logistic regression is a regression model to predict a binomially distributed response variable. Recent research has investigated the opportunity of combining logistic regression with decision tree learners. Following this idea, we propose a novel Logistic Model Tree induction system, SILoRT, which induces trees with two types of nodes: regression nodes, which perform only univariate logistic regression, and splitting nodes, which partition the feature space. The multiple regression model associated with a leaf is then built stepwise by combining univariate logistic regressions along the path from the root to the leaf. Internal regression nodes contribute to the definition of multiple models and have a global effect, while univariate regressions at leaves have only local effects. Experimental results are reported.

1 Introduction

In its original definition, logistic regression is a regression model for predicting the value of a binomially distributed response variable $Y = \{C_1, C_2\}$. Given a training set $D = \{(\boldsymbol{x}, y) \in \boldsymbol{X} \times Y \mid y = g(\boldsymbol{x})\}$ where $\boldsymbol{X}$ represents the search space spanned by m independent (or predictor) continuous variables X_i, a logistic regression model M is induced by generalizing observations in D in order to estimate the posterior class probability $P(C_i|x)$ that any unlabeled example $x \in \boldsymbol{X}$ belongs to C_i. Differently from the classical regression setting where the value of a (continuous) response variable is directly predicted, in logistic regression the response to be predicted is the probability that an example belongs to a given class. This probability can then be used for classification purposes.

Logistic regression was widely investigated in the literature for the classification task [3] [8] [5] [4]. The results of an empirical and formal comparison between logistic regression and decision trees [7] motivated the attempt of combining tree induction procedures with logistic regression. For example, Landwehr et al. [4] proposed a top-down fitting process to construct Logistic Model Trees. In this case, coefficients of a logistic regression model at leaves are constructed by exploiting the coefficients of logistic regression models constructed in the highest levels of the tree. Although correct, this approach considers only full regression models (i.e., regression models at leaves include all predictor variables) by ignoring that a regression model based on a subset of predictor variables may give more precise predictions than the model based on more variables [2]. This

A. An et al. (Eds.): ISMIS 2008, LNAI 4994, pp. 68–77, 2008.

depends on the fact that the variable subset selection avoids to poorly estimate regression coefficients in presence of two or more predictor variables linearly related to each other (collinearity) [6]. However, finding the best subset of variables while choosing the best split becomes too costly when applied to large datasets since the search procedure may require the computation of a high number of multiple regression models. Chan et al. [1] proposed to recursively partitioning the data and fitting a piecewise (multiple or univariate) logistic regression function in each partition. A user-defined parameter allows the user to choose between learning either multiple or univariate functions when constructing the logistic model to be associated with a node of the tree. In the case of a univariate logistic model, the collinearity problem is easily faced. In fact, the leaf model involves just a single predictor variable (univariate regression), but ignores all the others. Although its advantages in facing collinearity, the method proposed by Chan et al., as well as all the methods cited in this work, do not permit to capture the possible global effect of some predictor variable, that is, the case that the contribution of a predictor variable is equally shared by several models. Finally, Zeleis et al. [10] have recently proposed a model-based recursive partitioning which can be applied to various regression problems (linear regression and logistic regression), but, as in other cited methods, this partitioning is not able to discriminate between global and local effect of variables.

In this paper we propose a new top-down logistic model tree induction method, called SILoRT, which integrates the predictive phase and the splitting phase in the induction of a logistic model tree. Such logistic model trees include two types of nodes: regression nodes and split nodes [6]. The former are associated with univariate logistic regressions involving one continuous predictor variable, while the latter are associated with split tests. The multiple logistic model associated with each leaf is then built stepwise by combining all univariate logistic regressions reported along the path from the root to the leaf. This stepwise construction has several advantages. Firstly, it overcomes the computational problem of testing a large number of multiple linear regression models. Secondly, differently from original logistic regression formulation problem, it permits to consider both continuous and discrete variables. Thirdly, it solves the problem of collinearity since only the subset of variables selected with the regression nodes along the path from to root to the leaf is practically used to construct the logistic model to be associated with the leaf itself. Fourthly, it allows modeling phenomena where some variables have a global effect while others have only a local effect. Modeling such phenomena permits to obtain simpler model that can be easily understood by humans. A variable selected with a regression node at higher level of the tree has a global effect. In fact, the effect of the univariate logistic regression with this regression node is shared by all multiple models associated with the leaves of the sub-tree rooted in the regression node itself.

The paper is organized as follows. In the next Section, we briefly describe the stepwise construction of logistic model trees, while in Section 3 we describe the construction of the logistic regression model in SILoRT. Experimental results are commented in Section 4. Finally, conclusions and future works are drawn.

2 Stepwise Construction of Logistic Model Trees

The development of a tree structure is not only determined by a recursive partitioning procedure, but also by some intermediate prediction functions. This means that there are two types of nodes in the tree: regression nodes and splitting nodes. They pass down observations to their children in two different ways. For a splitting node t, only a subgroup of the $N(t)$ observations in t is passed to each child, and no change is made on the variables. For a regression node t, all the observations are passed down to its only child, but both the values of the response variable and the values of the (continuous) predictor variables not yet included in the model are transformed. The value of the response variable is transformed in order to take into account the error performed by the logistic regression function. Predictor variables not selected in the regression nodes along the path from the root to the current node are transformed in order to remove the linear effect of those variables already included in the model. Hence, each continuous predictor variable X_j not selected in a regression node is transformed in X'_j with $X'_j = X_j - (\alpha_0 + \alpha_1 X_i)$. X_i is the regression variable, α_0 and α_1 are intercept and slope estimated to model the straight-line regression between X_j and X_i ($\widehat{X_j} = \alpha_0 + \alpha_1 X_i$). α_0 and α_1 are computed according to the procedure in [2]. Thus, descendants of a regression node do operate on a modified dataset. This transformation is coherent with the stepwise procedure that is adopted in statistics to construct incrementally a multiple linear regression model: each time a new continuous predictor variable is added to the model its linear effect on remaining continuous variables has to be removed [2].

The validity of either a regression step or a splitting test on a variable X_i is based on two distinct evaluation measures, $\rho(X_i, Y)$ and $\sigma(X_i, Y)$ respectively. The variable X_i is of a continuous type in the former case, and of any type in the latter case. Both $\rho(X_i, Y)$ and $\sigma(X_i, Y)$ are error rates measures (i.e., percentage of misclassified cases), therefore they can be actually compared to choose between three different possibilities: (i) growing the model tree by adding a regression node t, (ii) growing the model tree by adding a split node t, (iii) stopping the tree growth at node t. The evaluation measure $\sigma(X_i, Y)$ is coherently defined on the basis of the multiple logistic regression model to be associated with each leaf. In the case of a split node, it is sufficient to consider the best univariate logistic regression associated to each leaf t_R (t_L), since regression nodes along the path from the root to t_R (t_L) already partially define a multiple logistic regression model. If X_i is continuous and α is a threshold value for X_i then $\sigma(X_i, Y)$ is defined as:

$$\sigma(X_i, Y) = \frac{N(t_L)}{N(t)} E(t_L) + \frac{N(t_R)}{N(t)} E(t_R) \tag{1}$$

where $N(t)$ is the number of cases reaching t, $N(t_L)$ ($N(t_R)$) is the number of cases passed down to the left (right) child, and $E(t_L)$ ($E(t_R)$) is the error rate of the left (right) child.

The error rate $E(t_L)(E(t_R))$ is computed by considering labeled (training) cases falling in t_L (t_R) and counting the cases whose class is different from

the class predicted by combining all logistic regression functions associated to regression nodes along the path from the root to t_L (t_R). More precisely:

$$E(t) = (1/N(t)) \sum_{\boldsymbol{x} \in t} d(y, \hat{y}). \quad (2)$$

where $\hat{y}$ is the class predicted for $\boldsymbol{x}$ and $d(y, \hat{y}) = 0$ if $y = \hat{y}$, 1 otherwise. The details on the construction of the logistic regression function at a node t are provided in Section 3.

Possible values of α are found by sorting the distinct values of X_i in the training set associated to t, then identifying one threshold between each pair of adjacent values. Therefore, if the cases in t have k distinct values for X_i, $k-1$ thresholds are considered. Obviously, the lower $\sigma(X_i, Y)$, the better the split.

If X_i is discrete, SILoRT partitions attribute values into two sets, the system starts with an empty set $LeftX = \emptyset$ and a full set $RightX = Sx$. It moves one element from $RightX$ to $LeftX$ such that the move results in a better split. The evaluation measure $\sigma(X_i, Y)$ is computed as in the case of continuous variables, therefore, a better split decreases $\sigma(X_i, Y)$. The process is iterated until there is no improvement in the splits.

The split selection criterion explained above can be improved to consider the special case of identical logistic regression model associated to both children (left and right). When this occurs, the straight-line regression associated to t is the same as that associated to both t_L and t_R, up to some statistically insignificant difference. In other words, the split is useless and can be filtered out from the set of alternatives. To check this special case, SILoRT compares the two regression lines associated to the children according to a statistical test for coincident regression lines [9].

Similarly to the splitting case, the evaluation of a regression step on a regression variable X_i is based on the error rate at node t: $\rho(X_i, Y) = E(t)$. $E(t)$ is computed as reported in (2). The choice between the best regression and the best split is performed by considering the following function $max\{\gamma \times max_i\{\sigma(X_i, Y)\}, max_i\{\rho(X_i, Y)\}\}$ where γ is a user defined parameter.

Three different stopping criteria are implemented in SILoRT. The first requires the number of cases in each node to be greater than a minimum value ($\sqrt{n}$). The second stops the induction process when all continuous predictor variables along the path from the root to the current node are selected in regression steps. The third stops the induction process when the predictive accuracy at the current node is 1 (i.e., error rate is 0).

3 Logistic Regression in SILoRT

SILoRT associates each node with a multiple logistic regression model that is constructed by combining all univariate logistic regressions associated with regression nodes along the path from the root to the node itself. Details on both the univariate logistic regression construction and the stepwise construction of multiple logistic regression functions in SILort are reported below.

3.1 Computing Univariate Logistic Regression Functions

A univariate logistic model on a (continuous) predictor variable X_i is the estimate of the posterior probability $P(Y|x_i)$ by means of the logit function:

$$P(C_1|x_i) = \frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}$$
$$P(C_0|x_i) = 1 - P(C_1|x_i) = \frac{1}{1+e^{\beta_0+\beta_1 x_i}}$$

Parameters $\beta = [\beta_0, \beta_1]^T$ are computed by maximizing the conditional log-likelihood of the class labels in the training dataset:

$$L(\beta) = \sum_{j=1..n} \widetilde{y}_i ln(P(C_1|x_{ij}, \beta)) + (1-\widetilde{y}_i) ln(P(C_0|x_{ij}, \beta)) \tag{3}$$

where $\widetilde{y}_i = 1$ if $y_i = C_1$, 0 otherwise and n is the number of examples.

The values β_0 and β_1 which maximize $L(\beta)$ are found by computing the first order partial derivative of $L(\beta)$ with respect to β and solving the system of equations:

$$\begin{cases} \frac{\partial L(\beta)}{\partial \beta_0} = 0 \\ \frac{\partial L(\beta)}{\partial \beta_1} = 0 \end{cases} \tag{4}$$

This system of equations can be solved by means of the Newton-Raphson algorithm. In particular, β is iteratively modified in order to reach the $L(\beta)$ zero. In the matrix representation,

$$\beta_{new} = \beta_{old} - \left(\frac{\partial^2 L(\beta)}{\partial \beta \partial \beta^T}\right)^{-1} \frac{\partial L(\beta)}{\partial \beta} \tag{5}$$

Operatively, β is computed according to Algorithm 1, where $minDist$ and $numIters$ are user defined thresholds used to stop the iteration.

3.2 Combining Univariate Logistic Regressions in a Stepwise Fashion

In order the combination of logistic regression functions, let us consider a simple example where training cases are described by the continuous predictor variables X_1 and X_2, while the response variable Y assumes values 0 and 1. A logistic model of Y on X_1 and X_2 is built stepwise by combining univariate logistic regressions with the regression nodes on X_1 and X_2, respectively.

We firstly derive parameters β_0 and β_1 of the univariate logistic regression of Y on X_1, such that:

$$\hat{Y} = \begin{cases} 1 & if \quad \frac{e^{\beta_0+\beta_1 X_1}}{1+e^{\beta_0+\beta_1 X_1}} > th \\ 0 & otherwise \end{cases} \tag{6}$$

β_0 and β_1 are computed according to the Algorithm 1. The fitted logistic model reported in Equation 6 may not predict Y exactly, but the error in predicting Y

Algorithm 1. Newton-Raphson application

$\beta \leftarrow [0,0]^T$; $numIters \leftarrow 0$;
$Q \leftarrow \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{i1} & x_{i2} & \dots & x_{in} \end{bmatrix}^T$;
$T \leftarrow [\widetilde{y}_1, \widetilde{y}_2, \dots, \widetilde{y}_n]^T$;
repeat
  $\beta_{new} \leftarrow \beta$; $numIters++$;
  $P \leftarrow [P(C_1|x_{i1},\beta), P(C_1|x_{i2},\beta), \dots, P(C_1|x_{in},\beta)]^T$;
  $\widetilde{X} \leftarrow \begin{bmatrix} P(C_1|x_{i1})P(C_0|x_{i1})[1,x_{i1}] \\ P(C_1|x_{i2})P(C_0|x_{i2})[1,x_{i2}] \\ \dots \\ P(C_1|x_{in})P(C_0|x_{in})[1,x_{in}] \end{bmatrix}$;
  $\beta_{new} \leftarrow \beta + (Q\widetilde{X})^{-1}Q(T-P)$;
until $||\beta_{new} - \beta||_2 > minDist$ OR $numIters > maxIters$
return β_{new}

may be reduced by adding the new variable X_2. Instead of starting from scratch and building a model with both X_1 and X_2, we exploit the stepwise procedure and now derive the slope and intercept of the straight-line regression to predict X_2 from X_1, that is:

$$\hat{X}_2 = \alpha_{2_0} + \alpha_{2_1} X_1. \tag{7}$$

Then we compute the residuals of both X_2 and Y as reported in (8)-(10).

$$X_2' = X_2 - (\alpha_{2_0} + \alpha_{2_l} X_i) \tag{8}$$

$$Y'_{FN} = \begin{cases} 1 & if \quad Y - \frac{e^{\beta_0+\beta_1 X_1}}{1+e^{\beta_0+\beta_1 X_1}} - th > 0 \\ 0 & otherwise \end{cases} \tag{9}$$

$$Y'_{FP} = \begin{cases} 1 & if \quad \frac{e^{\beta_0+\beta_1 X_1}}{1+e^{\beta_0+\beta_1 X_1}} - Y - th > 0 \\ 0 & otherwise \end{cases} \tag{10}$$

where th is a user defined threshold (default value 0.5) that identifies the "change of class" probability. Y'_{FN} are residuals for False Negative errors and Y'_{FP} are residuals for False Positive errors.

Both Y'_{FN} and Y'_{FP} are heuristically compared in order to choose the residual variable that minimizes the error rate on training cases. We denote this minimum with Y' and now compute the univariate logistic regression between Y' on X'_2. We obtain the logistic regression model:

$$\widehat{Y'} = \begin{cases} 1 & if \quad \frac{e^{\beta'_0+\beta'_1 X'_2}}{1+e^{\beta'_0+\beta'_1 X'_2}} > th \\ 0 & otherwise \end{cases} \tag{11}$$

and combining residuals, we have that

$$\hat{Y} = \begin{cases} 1 & if \quad \frac{e^{\beta_0+\beta_1 X_1}}{1+e^{\beta_0+\beta_1 X_1}} + s' \frac{e^{\beta'_0+\beta'_1 X'_2}}{1+e^{\beta'_0+\beta'_1 X'_2}} > th \\ 0 & otherwise \end{cases} \tag{12}$$

where s'=1 if we are performing a logistic regression on Y'_{FN}, s'=-1 if we are performing a logistic regression on Y'_{FP}.

A generalization of this procedure is that adopted by SILoRT to construct the multiple logistic regression model at a node t in a stepwise fashion. Logistic univariate regressions to be combined are those constructed in regression nodes along the path from the root to the node t. Parameters of both logistic and straight-line regressions at each node of the tree are estimated by processing only the training cases falling in the node itself. Intuitively, the stepwise construction of a logistic model tree is motivated by the need of reducing the classification error performed in regression nodes at higher levels of the tree by adding new predictor variables to the model by means of the regression nodes (or leaves) at lower levels of the tree.

4 Evaluation

In order to evaluate performances of SILoRT, we have commented an illustrative example and reported an empirical evaluation.

4.1 An Illustrative Example

Let us consider training data described by two continuous predictor variables X_1 and X_2 and a binary response variable Y, that is:

$X1$	1	2	3	3	3	4	5	6	7	8	20
$X2$	1	2	9	10	11	8	7	6	5	4	3
Y	1	0	0	0	0	0	0	0	0	0	1

SILoRT ($\gamma = 0.9$, $th = 0.5$) chooses the regression node on X_1 as root of the tree (Figure 1). This means that a global effect of X_1 on Y has been observed. The univariate logistic regression on $X1$ is the following:

$$\widehat{Y} = \begin{cases} 1 & if \quad \frac{e^{-2.797+0.191*X1}}{1+e^{-2.797+0.191*X1}} > th \\ 0 & otherwise \end{cases}$$

and the linear regression of $X2$ on $X1$ is: $\widehat{X2} = (6.97 - 0.17 * X1)$.

The parameters of the logistic regression function are determined as reported in Algorithm 1. Training data are then transformed in order to remove the effect of performed (logistic) regression and they are passed down to the unique child. In particular, $X2$ is replaced with residual $X2'$ while response variable Y is replaced by a pair of variables including the residuals of False Positive and False Negative errors.

$X2'$	-5.8	-4.6	2.5	3.5	4.5	1.7	0.9	0.1	-0.8	-1.6	-0.5
Y'_{FN}	1	0	0	0	0	0	0	0	0	0	0
Y'_{FP}	0	0	0	0	0	0	0	0	0	0	0

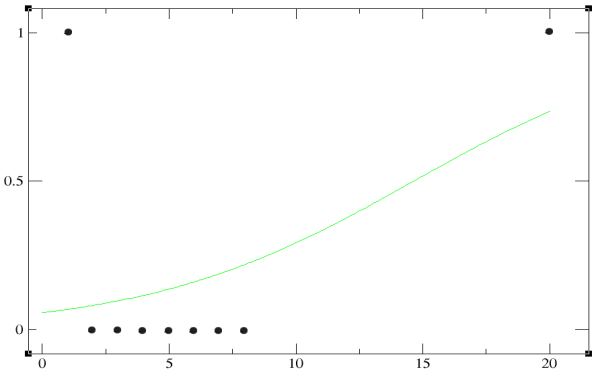

Fig. 1. Regression on $X1$

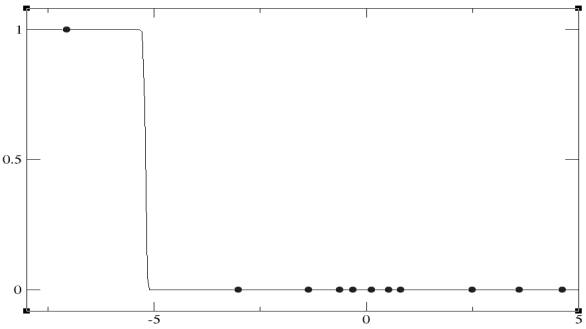

Fig. 2. Regression on $X2'$

SILoRT chooses to compute the logistic regression of Y'_{FN} from $X2'$ and obtains the logistic regression model (Figure 2):

$$\widehat{Y'_{FN}} = \begin{cases} 1 & if \quad \frac{e^{-330.4-63.4*X2'}}{1+e^{-330.4-63.4*X2'}} > th \\ 0 & otherwise \end{cases}$$

This logistic regression correctly predicts all training cases (error rate=0). If SILoRT chose to regress Y'_{FP}, error rate on the training set would be 9%.

Since there are no other continuous variables, SILoRT stops the search and returns a regression tree with two regression nodes (one of which is a leaf). This model is relatively simple and shows the global effect of $X1$ and $X2$ on Y.

4.2 Evaluation on Real World Datasets

SILoRT has been empirically evaluated on six datasets taken from the UCI Machine Learning Repository (http://www.ics.uci.edu/ mlearn/MLRepository.html) Each

Table 1. Datasets used in SILoRT evaluation

Dataset	#Cases	#Attributes	#Cont. Attributes
Breast-cancer-wisconsin	683	10	10
Ionosphere	351	34	34
Iris	100	4	4
Pima-indians-diabetes	768	8	8
Storm	3664	4	3
Wdbc	569	31	31

Table 2. SILoRT evaluation (average accuracy)

Dataset	SILoRT	Baseline
Breast-cancer-wisconsin	93.82	92.22
Ionosphere	85.09	81.76
Iris	91	91
Pima-indians-diabetes	69.02	74.35
Storm	79.29	69.38
Wdbc	93.15	88.74

Table 3. SILoRT: model complexity

Dataset	Avg. #Leaves	Avg. max tree depth
Breast-cancer-wisconsin	13.5	6.5
Ionosphere	12.7	6.56
Iris	4.3	3.76
Pima-indians-diabetes	40.9	12.44
Storm	133.1	35.64
Wdbc	10.5	8.46

data dataset is analyzed by means of a 10-fold cross-validation that is, the dataset was divided into ten *folds* and then, for every fold, SILoRT was trained on the remaining folds and tested on it. Main characteristics of used datasets are reported in Table 1. Results are obtained with the following parameters values $\gamma = 1, th = 0.5$, $minDist = 0.001, numIters = 400$.

As baseline of our experiments, we considered the simple classifier whose classification model is a univariate logistic regression model (see section 3.1). Results reported in Table 2 show a clear advantage of SILoRT with respect to the baseline in most of cases. This result is not so clear in case of few continuous attributes when SILoRt seems to suffer from overfitting problems. This is confirmed by the relative complexity of induced models (see Table 3). On the contrary, the system shows good results in case of discrete attributes. This result was somehow expected since the baseline algorithm, as original approaches for logistic regression, does not consider discrete attributes.

5 Conclusions

In this paper, we propose a new Logistic Model Trres induction method, SILoRT, whose induced model is characterized by two types of nodes: logistic regression nodes and splitting nodes. Main peculiarities of SILoRT are in the capability to solve collinearity problems without additional computational costs and in the capability of facing the problem of modeling phenomena, where some variables have a global effect while others have only a local effect. This permits to obtain models that can be easily understood by humans.

Similarly to many decision tree induction algorithms, SILoRT may generate model trees that overfit training data. Therefore, a future research direction is the a posteriori simplification of model trees with both regression nodes and splitting nodes. For future work, we also intend to evaluate the opportunity of considering discrete variables in regression models, empirically compare SILoRT with other classifiers and extend experiments by considering artificial data.

Acknowledgments

This work is supported in partial fulfillment of the research objectives of "FAR" project "Laboratorio di bioinformatica per la biodiversità molecolare".

References

1. Chan, K.Y., Loh, W.Y.: Lotus: An algorithm for building accurate and comprehensible logistic regression trees. Journal of Computational and Graphical Statistic 13(4), 826–852 (2004)
2. Draper, N.R., Smith, H.: Applied regression analysis. John Wiley & Sons, Chichester (1982)
3. Hosmer, D.W., Lemeshow, S.: Applied Logistic Regression. Wiley, New York (1989)
4. Landwehr, N., Hall, M., Frank, E.: Logistic model trees 95(1-2), 161–205 (2005)
5. le Cessie, S., van Houwelingen, J.: Ridge estimators in logistic regression. Applied Statistics 41(1), 191–201 (1992)
6. Malerba, D., Esposito, F., Ceci, M., Appice, A.: Top down induction of model trees with regression and splitting nodes. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(5), 612–625 (2004)
7. Perlich, C., Provost, F., Simonoff, J.S.: Tree induction vs. logistic regression: a learning-curve analysis. J. Mach. Learn. Res. 4, 211–255 (2003)
8. Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S-PLUS, 3rd edn. Springer, New York (1999)
9. Weisberg, S.: Applied regression analysis. Wiley, New York (1985)
10. Zeileis, A., Hothorn, T., Hornik, K.: A model-based recursive partitioning. Journal of Computational and Graphical Statistics (2008)