Model-based Recursive Partitioning

Achim Zeileis
Wirtschaftsuniversität Wien

Torsten Hothorn
Friedrich-Alexander-Universität
Erlangen-Nürnberg

Kurt Hornik
Wirtschaftsuniversität Wien

Abstract

Recursive partitioning is embedded into the general and well-established class of parametric models that can be fitted using M-type estimators (including maximum likelihood). An algorithm for model-based recursive partitioning is suggested for which the basic steps are: (1) fit a parametric model to a data set, (2) test for parameter instability over a set of partitioning variables, (3) if there is some overall parameter instability, split the model with respect to the variable associated with the highest instability, (4) repeat the procedure in each of the daughter nodes. The algorithm yields a partitioned (or segmented) parametric model that can be effectively visualized and that subject-matter scientists are used to analyze and interpret.

Keywords: change points, maximum likelihood, parameter instability, recursive partitioning.

1. Introduction

Since the appearance of the first tree-structured regression analysis (Automated Interaction Detection, Morgan and Sonquist 1963), virtually every publication in this field highlights two features of trees: 1. interpretability—enhanced by visualizations of the fitted decision trees—and 2. predictive power in non-linear regression relationships. The latter is of diminishing importance because modern approaches to predictive modeling such as boosting (e.g., simple L_2 boosting by Bühlmann and Yu 2003), random forests (Breiman 2001) or support vector machines (Vapnik 1996) are often found to be superior to trees in purely predictive settings (e.g., Meyer, Leisch, and Hornik 2003). However, a simple graphical representation of a complex regression problem is still very valuable, probably increasingly so.

In the last decade, the incorporation of (simple) parametric models into trees has been receiving increased interest. Research in this direction was mainly motivated by the fact that constant fits in each node tend to produce large and thus hard to interpret trees (see e.g., Chan and Loh 2004). Several algorithms have been suggested both in the statistical and machine learning communities that attach parametric models to terminal nodes or employ linear combinations to obtain splits in inner nodes. In machine learning, such approaches are known as *hybrid*, *model* or *functional trees* (Gama 2004) with M5 (Quinlan 1993) being the most prominent representative. The key developments in statistics are due to Wei-Yin Loh and his coworkers. GUIDE (Loh 2002), CRUISE (Kim and Loh 2001) and LOTUS (Chan and Loh 2004) attach parametric models to terminal nodes, and Choi, Ahn, and Chen (2005) suggest an extension to count data. Some of these algorithms (in particular CRUISE or QUEST, Loh and Shih 1997) additionally allow to employ parametric models to obtain splits in inner nodes. Furthermore, maximum likelihood trees (Su, Wang, and Fan 2004) embed regression trees with a constant fit in each terminal node into maximum likelihood estimation.

Building on these ideas, we carry the integration of parametric models into trees one step further and provide a rigorous theoretical foundation by introducing a new unified framework that embeds recursive partitioning into statistical model estimation and variable selection. Within this framework, a segmented parametric model is fitted by computing a tree in which every leaf is associated with a fitted model such as, e.g., a maximum likelihood model or a linear regression. The model's objective function is used for estimating the parameters and the split points, the corresponding model scores are tested for parameter instability in each node to assess which variable should be used for partitioning. The benefits of employing this approach are: The objective function used for parameter estimation is also used for partitioning (testing *and* split point estimation). The recursive partitioning allows for modeling of non-linear relationships and automated detection of interactions among the explanatory variables. The statistical formulation of the algorithm ensures the validity of interpretations drawn from the depicted model. Moreover, the use of well-known parametric models provides subject-matter scientists with a segmented model that they are used to analyze and interpret.

The remainder of the paper is organized as follows: Section 2 establishes the class of models the framework is based on before Section 3 describes the suggested model-based recursive partitioning algorithm in detail. Section 4 provides a set of illustrations and applications along with benchmark comparisons with other tree-based algorithms. Section 5 concludes the paper with a summary and some final remarks.

2. Segmented models

Consider a parametric model $\mathcal{M}(Y, \theta)$ with (possibly vector-valued) observations $Y \in \mathcal{Y}$ and a k-dimensional vector of parameters $\theta \in \Theta$. Given n observations Y_i $(i = 1, \dots, n)$ the model can be fitted by minimizing some objective function $\Psi(Y, \theta)$ yielding the parameter estimate $\hat{\theta}$

$$\hat{\theta} \quad = \quad \underset{\theta \in \Theta}{\operatorname{argmin}} \sum_{i=1}^{n} \Psi(Y_i, \theta). \tag{1}$$

Estimators of this type include various well-known estimation techniques, the most popular being ordinary least squares (OLS) or maximum likelihood (ML) among other M-type estimators. In the case of OLS, Ψ is typically the error sum of squares and, in the case of ML, it is the negative log-likelihood. In the latter case, it could be the full likelihood of the variable Y or the conditional likelihood if Y can be split into dependent and explanatory variables $Y = (y, x)^\top$.

Example: (Multivariate) normal distribution. The observations Y are normally distributed with mean μ and covariance matrix Σ: $Y \sim \mathcal{N}(\mu, \Sigma)$ with the combined parameter vector $\theta = (\mu, \Sigma)$.

Example: Generalized linear model (GLM). The observations can be split into a dependent variable y and covariates or regressors x, i.e., $Y = (y, x)^\top$. The model equation is $g(\mathsf{E}(y)) = x^\top \theta$ where y has a pre-specified exponential family distribution, $g(\cdot)$ is a known link function and θ are the regression coefficients.

In many situations, it is unreasonable to assume that a single global model $\mathcal{M}(Y, \theta)$ fits all n observations well. But it might be possible to partition the observations with respect to some covariates such that a well-fitting model can be found in each cell of the partition. In such a situation, we can use a recursive partitioning approach based on ℓ partitioning variables $Z_j \in \mathcal{Z}_j$ $(j = 1, \dots, \ell)$ to adaptively find a good approximation of this partition.

More formally, we assume that a partition $\{\mathcal{B}_b\}_{b=1,\dots,B}$ of the space $\mathcal{Z} = \mathcal{Z}_1 \times \dots \times \mathcal{Z}_\ell$ exists with B cells (or segments) such that in each cell $\mathcal{B}_b$ a model $\mathcal{M}(Y, \theta_b)$ with a cell-specific parameter θ_b holds. We denote this segmented model by $\mathcal{M}_\mathcal{B}(Y, \theta)$ where θ is now the full combined parameter $\theta = (\theta_1, \dots, \theta_B)^\top$.

Special cases of such segmented models are classification and regression trees where many partitioning variables Z_j but only very simple models $\mathcal{M}$ are used, and structural break models that find partitions with respect to time.

Example: For regression trees a simple model $\mathcal{M}$ is chosen: the parameter θ describes the mean of the univariate observations Y_i and is estimated by OLS (or equivalently ML in a normal model with the variance treated as a nuisance parameter). The variables Z_j are the regressors considered for partitioning.

Example: In change point or structural change analysis, typically a linear regression model with $Y_i = (y_i, x_i)^\top$ and regression coefficients θ is segmented with respect to only a single variable Z_1 (i.e., $\ell = 1$) which is usually time.

Given the correct partition $\{\mathcal{B}_b\}$ the estimation of the parameters θ that minimize the corresponding global objective function $\sum_{b=1}^{B} \sum_{i \in I_b} \Psi(Y_i, \theta_b)$ can be easily achieved by computing the locally optimal parameter estimates $\hat{\theta}_b$ in each segment $\mathcal{B}_b$ (with corresponding indexes I_b). But if the partition $\{\mathcal{B}_b\}$ is unknown, minimization of Ψ is more complicated, even if the number of segments B is fixed: If there is more than one partitioning variable ($\ell > 1$), the number of potential partitions quickly becomes too large for an exhaustive search. If, in addition, the number of segments B is unknown, the problem becomes even more severe—at least if trivial partitions such as the partition, where each observation is its own segment, are excluded, e.g., by requiring some minimum segment size. Furthermore, in this case, some means should be taken to avoid overfitting by increasing B.

In short, determining the optimal partition (with respect to Ψ) is difficult, even for fixed B. However, if there is only $\ell = 1$ partitioning variable, the optimal split(s) can be found easily: both the statistics and econometrics literature on change point and structural change analysis discuss various algorithms for segmenting models over a single variable, typically time. To exploit this methodology for finding a partition close to the optimal one in $\ell > 1$ dimensions, we suggest a greedy forward search where the objective function Ψ can at least be optimized locally in each step. A detailed description of this algorithm is given in the next section.

3. The recursive partitioning algorithm

The basic idea is that each node is associated with a single model. To assess whether splitting of the node is necessary a fluctuation test for parameter instability is performed. If there is significant instability with respect to any of the partitioning variables Z_j, split the node into B locally optimal segments and repeat the procedure. If no more significant instabilities can be found, the recursion stops and returns a tree where each terminal node (or leaf) is associated with a model of type $\mathcal{M}(Y, \theta)$. More precisely, the steps of the algorithm are

1. Fit the model once to all observations in the current node by estimating $\hat{\theta}$ via minimization of the objective function Ψ.
2. Assess whether the parameter estimates are stable with respect to every ordering $Z_1, \dots, Z_\ell$. If there is some overall instability, select the variable Z_j associated with the highest parameter instability, otherwise stop.
3. Compute the split point(s) that locally optimize Ψ, either for a fixed or adaptively chosen number of splits.
4. Split the node into daughter nodes and repeat the procedure.

The details for steps 1–3 are specified in the following. To keep the notation simple, the dependence on the current segment is suppressed and the symbols established for the global model are used, i.e., n for the number of observations in the current node, $\hat{\theta}$ for the associated parameter estimate and B for the number of daughter nodes chosen.

3.1. Parameter estimation

This step of the algorithm is common practice: Under mild regularity conditions (see e.g., White 1994), it can be shown that the estimate $\hat{\theta}$ defined by Equation 1 can also be computed by solving the first order conditions

$$\sum_{i=1}^{n} \psi(Y_i, \hat{\theta}) \quad = \quad 0, \tag{2}$$

where

$$\psi(Y,\theta) \quad = \quad \frac{\partial \Psi(Y,\theta)}{\partial \theta} \tag{3}$$

is the score function or estimating function corresponding to $\Psi(Y,\theta)$. Analytical closed form solutions are available only in certain special cases, but for many models of interest well-established fitting algorithms for computing $\hat{\theta}$ are available. The score function evaluated at the estimated parameters $\hat{\psi}_i = \psi(Y_i, \hat{\theta})$ is then inspected for systematic deviations from its mean 0 in the next section.

3.2. Testing for parameter instability

The task in this step of the algorithm is to find out whether the parameters of the fitted model are stable over each particular ordering implied by the partitioning variables Z_j or whether splitting the sample with respect to one of the Z_j might capture instabilities in the parameters and thus improve the fit. To assess the parameter instability, a natural idea is to check whether the scores $\hat{\psi}_i$ fluctuate randomly around their mean 0 or exhibit systematic deviations from 0 over Z_j. These deviations can be captured by the empirical fluctuation process

$$W_j(t) \quad = \quad \hat{J}^{-1/2} n^{-1/2} \sum_{i=1}^{\lfloor nt \rfloor} \hat{\psi}_{\sigma(Z_{ij})} \qquad (0 \le t \le 1) \tag{4}$$

where $\sigma(Z_{ij})$ is the ordering permutation which gives the antirank of the observation Z_{ij} in the vector $Z_j = (Z_{1j}, \dots, Z_{nj})^\top$. Thus, $W_j(t)$ is simply the partial sum process of the scores ordered by the variable Z_j, scaled by the number of observations n and a suitable estimate $\hat{J}$ of the covariance matrix $\mathsf{COV}(\psi(Y,\hat{\theta}))$, e.g., $\hat{J} = n^{-1} \sum_{i=1}^{n} \psi(Y_i,\hat{\theta})\psi(Y_i,\hat{\theta})^\top$, but other robust estimators such as HC (heteroskedasticity consistent) and HAC (heteroskedasticity and autocorrelation consistent) estimators are also applicable. This empirical fluctuation process is governed by a functional central limit theorem (Zeileis and Hornik 2003) under the null hypothesis of parameter stability: it converges to a Brownian bridge W^0. A test statistic can be derived by applying a scalar functional $\lambda(\cdot)$ capturing the fluctuation in the empirical process to the fluctuation process $\lambda(W_j(\cdot))$ and the corresponding limiting distribution is just the same functional (or its asymptotic counterpart) applied to the limiting process $\lambda(W^0(\cdot))$.

This very general framework for testing parameter stability is called generalized M-fluctuation test and has been established by Zeileis and Hornik (2003). It has been shown to encompass a large number of structural change tests suggested both in the econometrics and statistics literature, including OLS-based CUSUM and MOSUM tests (Ploberger and Krämer 1992; Chu, Hornik, and Kuan 1995), score-based tests (Nyblom 1989; Hjort and Koning 2002) and statistics based on lagrange multiplier statistics (Andrews 1993; Andrews and Ploberger 1994)—an overview is given in Zeileis (2005). In principle, any of the tests from this framework could be used in the recursive partitioning algorithm, but two different test statistics seem to be particularly attractive for assessing numerical and categorical partitioning variables Z_j respectively.

Assessing numerical variables: To capture the instabilities over a numerical variable Z_j, the following functional is most intuitive:

$$\lambda_{\sup LM}(W_j) \quad = \quad \max_{i=\underline{\imath},\dots,\overline{\imath}} \left(\frac{i}{n} \cdot \frac{n-i}{n}\right)^{-1} \left|\left| W_j\left(\frac{i}{n}\right)\right|\right|_2^2, \tag{5}$$

which is the maximum of the squared L_2 norm of the empirical fluctuation process scaled by its variance function. This is the supLM statistic of Andrews (1993) which can be interpreted as the supremum of LM statistics against a single change point alternative where the potential change point is shifted over the interval $[\underline{\imath}, \overline{\imath}]$ that is typically defined by requiring some minimal segment size $\underline{\imath}$ and then $\overline{\imath} = n - \underline{\imath}$. This test statistic is asymptotically equivalent to the supremum of likelihood-ratio (or Chow) statistics (Chow 1960) but has the advantage that the model has to be fitted only once under the null hypothesis of parameter stability and not under the alternative for each conceivable change point. The limiting distribution of the supLM statistic is given by the supremum of a squared, k-dimensional tied-down Bessel process $\sup_t (t(1-t))^{-1} ||W^0(t)||_2^2$ from which the corresponding p value p_j can be computed (Hansen 1997).

Recently, Potts and Sammut (2005) have also pointed out that parameter instability tests can be employed for growing linear regression trees. However, our approach makes the evaluation of all conceivable change points feasible (and not only a fixed number, such as 5), uses the correct limiting distribution (rather than a crude Bonferroni approximation) and can be applied to a more general class of models.

Assessing categorical variables: To capture the instability with respect to a categorical variable Z_j with C different levels or categories, a different statistic is required because, by definition, Z_j has ties and hence a total ordering of the observations is not possible. The most natural statistic, which is insensitive to the ordering of the C levels and the ordering of observations within each level, is given by

$$\lambda_{\chi^2}(W_j) \quad = \quad \sum_{c=1}^{C} \frac{|I_c|}{n}^{-1} \left|\left| \Delta_{I_c} W_j \left(\frac{i}{n} \right) \right|\right|_2^2 \tag{6}$$

where $\Delta_{I_c} W_j$ is the increment of the empirical fluctuation process over the observations in category $c = 1, \dots, C$ (i.e., essentially the sum of the scores in category c). The test statistic is then the weighted sum of the squared L_2 norm of the increments which has an asymptotic χ^2 distribution with $k \cdot (C-1)$ degrees of freedom from which the corresponding p value p_j can be computed (Hjort and Koning 2002).

The advantage of using this approach based on the empirical fluctuation processes from Equation 4 with the functionals from Equations 5 and 6 is that the parameter estimates and corresponding score functions just have to be computed once in a node. For performing the parameter instability tests, the scores just have to be reordered and aggregated to a scalar test statistic each time.

To test whether there is some overall instability in the current node, it just has to be checked whether the minimal p value $\min_{j=1,\dots,\ell} p_j$ falls below a pre-specified significance level α, that is typically corrected for multiple testing. If this is the case, the variable Z_{j^*} associated with the minimal p value is chosen for splitting the model in the next step of the algorithm.

3.3. Splitting

In this step of the algorithm the fitted model has to be split with respect to the variable Z_{j^*} into a segmented model with B segments where B can either be fixed or determined adaptively. For a fixed number of splits, two rival segmentations can be compared easily by comparing the segmented objective function $\sum_{b=1}^{B} \sum_{i \in I_b} \Psi(Y_i, \theta_b)$. Performing an exhaustive search over all conceivable partitions with B segments is guaranteed to find the optimal partition but might be burdensome, so several search methods are briefly discussed for numerical and categorical partitioning variables respectively.

Splitting numerical variables: Exhaustive search for a split into $B = 2$ segments is feasible in $O(n)$ operations. For $B > 2$, when an exhaustive search would be of order $O(n^{B-1})$, the optimal partition can be found using a dynamic programming approach of order $O(n^2)$. This is an application of Bellman's principle and has been discussed in several places in the statistics and econometrics literature on change point and structural change analysis (see e.g., Hawkins 2001; Bai and Perron 2003; Zeileis, Kleiber, Krämer, and Hornik 2003, among others). Alternatively, iterative algorithms can be used that are known to converge to the optimal solution (e.g., Muggeo 2003). If B is not fixed, but should be chosen adaptively, various methods are available (see e.g., Bai and Perron 2003; O'Brien 2004). In particular, information criteria can be used if the parameters are estimated by ML.

Splitting categorical variables: For categorical variables, the number of segments can not be larger than the number of categories $B \leq C$. Two simple approaches would be either to always split into all $B = C$ possible levels or alternatively to always split into the minimal number of $B = 2$ segments. In this case, the search for the optimal partition is of order $O(2^{C-1})$. For ordinal variables, it also makes sense to just split in the ordering of the levels, so that the search for a binary split is only of order $O(C)$. Again, information criteria could be an option to adaptively determine the number of splits, although this is less intuitive than for numerical variables.

In summary, two plausible strategies would be either to always use binary splits, i.e., use a fixed $B = 2$, or to determine B adaptively for numerical variables while always using $B = C$ for categorical variables. In Section 4 below, we adopt the former strategy of binary splits.

This concludes one iteration of the recursive partitioning algorithm and steps 1–3 are carried out again in each of the B daughter nodes until no significant instability is detected in step 2.

4. Illustrations and applications

To illustrate how model-based recursive partitioning can be used in practice, the general framework from the previous section is applied to various regression problems (linear regression, logistic regression, and survival regression). Four different data sets are analyzed in the following way: In a first step, the recursively partitioned model is fitted to the data and visualized for explanatory analysis, emphasizing that the algorithm can be used to build intelligible local models by automated interaction detection. In a second step, the performance of the algorithm is compared with other tree-based algorithms in two different respects: prediction and complexity. Comparing predictive performance of different learning algorithms is established practice—for (model-based) recursive partitioning comparing the model complexity (i.e., the number of splits and estimated coefficients) is equally important. As argued above, the strength of single tree-based classifiers is not so much predictive power alone, but that the algorithms are able to build interpretable models. Clearly, more parsimonious models are easier to interpret and hence are to be preferred (among those with comparable predictive performance).

For the linear regression applications, the model-based (MOB) recursive partitioning algorithm introduced here is compared to other algorithms previously suggested in the literature: GUIDE (Loh 2002) and M5' (Wang and Witten 1997), which is a rational reconstruction of M5 (Quinlan 1992), as linear model trees; as well as CART (classification and regression trees, Breiman, Friedman, Olshen, and Stone 1984) and conditional inference trees (CTree, Hothorn, Hornik, and Zeileis 2006a) as trees with constant models in the nodes. The logistic regression-based MOB trees are compared with logistic model trees (LMT, Landwehr, Hall, and Frank 2005) as well as various tree-based algorithms with constant models in the nodes: QUEST (Loh and Shih 1997), the J4.8 implementation of C4.5 (Quinlan 1993), CART and CTree.

All benchmark comparisons are carried out in the framework of Hothorn, Leisch, Zeileis, and Hornik (2005) based on 250 bootstrap replications and employing the root mean squared error (RMSE) and misclassification rate on the out-of-bag (OOB) samples as predictive performance measure and the number of estimated parameters (splits and coefficients) as complexity measure. The median performances[1] on the bootstrap replications are reported in tabular form, simultaneous confidence intervals for performance differences (obtained by treatment contrasts with MOB as the reference category) are visualized. In addition, the tables contain the obvious complexity and prediction performance measures (RMSE or misclassification) on the original data set as an additional reference information (although the obvious prediction measures obviously represent no honest estimators).

Most computations have been carried out in the R system for statistical computing (R Development Core Team 2005), in particular using the packages **party** (Hothorn, Zeileis, and Hornik 2006b), providing implementations of MOB and CTree, **rpart** (Therneau and Atkinson 1997), implementing CART, and **RWeka** (Hornik, Zeileis, Hothorn, and Buchta 2006), the R interface to

[1] The median rather than the mean is used for two reasons: First, to account for the skewness of the performance distributions, and second due to the many bindings in the complexity measure for the model-based tree algorithms (MOB, GUIDE, M5', LMT).

Weka (Witten and Frank 2005) containing implementations of M5', LMT and J4.8. For GUIDE and QUEST, the binaries distributed at `http://www.stat.wisc.edu/~loh/` were used.

4.1. Demand for economic journals

Journal pricing is a topic that stirred considerable interest in the economics literature in recent years, see Bergstrom (2001) and his journal pricing Web page `http://www.econ.ucsb.edu/~tedb/Journals/jpricing.html` for further informations on this discussion. Using data collected by T. Bergstrom for $n = 180$ economic journals, Stock and Watson (2003) fit a demand equation by OLS for the number of library subscriptions explained by the price per citation (both in logs). In their analysis, they find that this simple linear regression can be improved by including further variables such as age, number of characters and interactions of age and price into the model with no clear solution what is the best way of incorporating these further variables.

This is where we set out with an analysis by means of model-based recursive partitioning. The model to be partitioned is a linear regression for number of library subscriptions by price per citation in log-log specification (i.e., with $k = 2$ coefficients). The $\ell = 5$ partitioning variables are the raw price and number of citations, the age, number of characters and a factor indicating whether the journal is associated with a society or not. Thus, we use a standard model whose specification is driven by economic knowledge and try to partition it with respect to further variables whose influence is not clear in advance. Note that whereas the selection of appropriate transformations is crucial for the modeling variables, monotonous transformations of the partitioning variables have no influence on the fitting process. For testing, a Bonferroni-corrected significance level of $\alpha = 0.05$ and a minimal segment size of $\underline{i} = 10$ is used.

The resulting linear regression-based tree for the economic journals data is depicted in Figure 1 employing scatter plots with fitted regression lines in the leaves. In the fitting process, a global model for all observations is estimated (yielding a price elasticity of -0.53) and assessed by the parameter instability tests. A highly significant instability is found only with respect to age (with a Bonferroni-adjusted p value of $p < 0.001$) which is subsequently used for splitting, leading to an optimal split at age 18. For the 53 young journals a much higher price elasticity of -0.6 is found than for the 127 older journals with a price elasticity of -0.4. No further parameter instabilities with respect to the partitioning variables can be detected (all p values are greater than 90%) and hence the algorithm stops. Table 1 reports the RMSE on the original data and the model complexity of 5 (2 times $k = 2$ coefficients plus $2 - 1$ splits).

Table 1 and Figure 2 provide the results of the benchmark comparison on 250 bootstrap samples. The MOB trees have the lowest median RMSE and complexity, the simultaneous confidence intervals show that the differences compared to GUIDE are non-significant in both cases and significant compared to all other models. While the trees with constant fits are clearly outperformed with respect to RMSE, M5' is still quite close in terms of RMSE but requires a substantially larger number of parameters to achieve this predictive performance.

	RMSE		Number of parameters	
	Bootstrap	Original	Bootstrap	Original
MOB	0.730	0.654	8	5
GUIDE	0.734	0.606	13	13
M5'	0.752	0.625	22	19
CTree	0.806	0.710	11	9
RPart	0.804	0.669	17	9

Table 1: Performance comparison for economic journals data: prediction error is compared by median root mean squared error (RMSE) on 250 bootstrap samples and obvious RMSE on the original data set; complexity is compared by (median) number of estimated parameters.

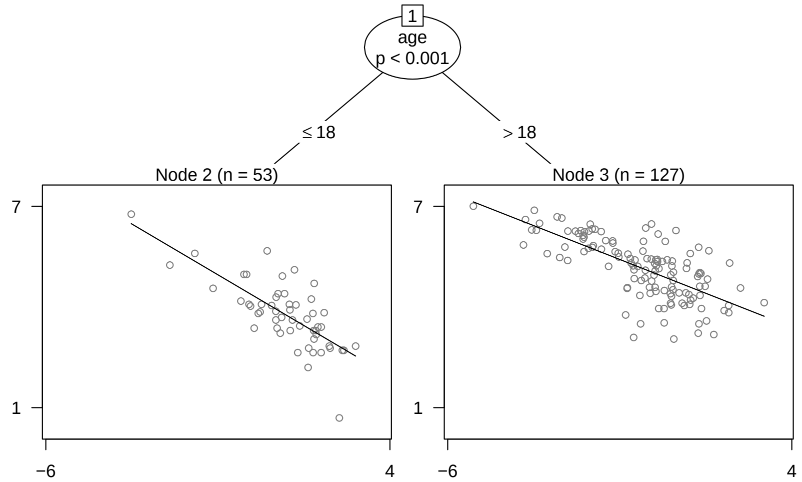

Figure 1: Linear-regression-based tree for the economic journals data. The plots in the leaves depict library subsriptions by price per citation (both in logs).

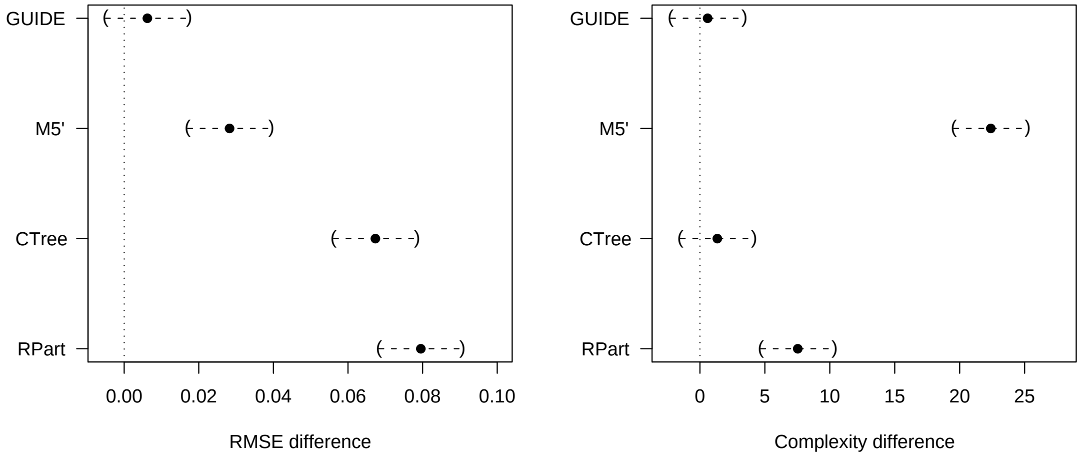

Figure 2: Performance comparison for economic journals data: prediction error is compared by RMSE differences, complexity by difference in number of estimated parameters (coefficients and split points).

4.2. Boston housing data

Since the analysis by Breiman and Friedman (1985), the Boston housing data are a popular and well-investigated empirical basis for illustrating non-linear regression methods both in machine learning and statistics (see Gama 2004; Samarov, Spokoiny, and Vial 2005, for two recent examples) and we follow these examples by segmenting a bivariate linear regression model for the house values.

The data set provides $n = 506$ observations of the median value of owner-occupied homes in Boston (in USD 1000) along with 14 covariates including in particular the number of rooms per dwelling (rm) and the percentage of lower status of the population (lstat). A segment-wise linear relationship between the value and these two variables is very intuitive, whereas the shape of the influence of the remaining covariates is rather unclear and hence should be learned from the data. Therefore, a linear regression model for median value explained by $(\text{rm})^2$ and $\log(\text{lstat})$ with $k = 3$ regression coefficients is employed and partitioned with respect to all $\ell = 11$ remaining variables. As argued above, choosing appropriate transformations of the modeling variables is important to obtain a well-fitting model in each segment and we follow in our choice the recommendations of Breiman and Friedman (1985). The model is estimated by OLS, the instability is assessed using a Bonferroni-corrected significance level of $\alpha = 0.05$ and the nodes are split with a required minimal segment size of $\underline{i} = 40$.

The resulting model-based tree is depicted in Figure 3 which shows partial scatter plots along with the fitted values in the terminal nodes. It can be seen that in the nodes 3, 6 and 7 the increase of value with the number of rooms dominates the picture (lower panel) whereas in node 9 the decrease with the lower status population percentage (upper panel) is more pronounced. Splits are performed in the variables tax (property-tax rate) and ptratio (pupil-teacher ratio). As reported in Table 2, the model has $5 \cdot 3$ regression coefficients after estimating $5 - 1$ splits, giving a total of 19 estimated parameters.

The results of the benchmark comparison in Table 2 and Figure 4 show that the MOB trees perform significantly better on this data set than the other tree-based algorithms. The algorithm with the most comparable predictive performance, M5', is here clearly inferior concerning its interpretability, requiring on average more than 12 times as many parameters.

	RMSE		Number of parameters	
	Bootstrap	Original	Bootstrap	Original
MOB	3.975	3.469	27	19
GUIDE	4.378	4.137	13	13
M5'	4.058	2.482	348	321
CTree	4.607	3.428	43	37
RPart	4.838	4.193	17	13

Table 2: Performance comparison for Boston housing data: prediction error is compared by RMSE on 250 bootstrap samples and obvious RMSE on the original data set; complexity is compared by (median) number of estimated parameters.

4.3. Pima Indians diabetes data

Another popular data set for comparing new classifiers is the Pima Indians diabetes data which is—just as the Boston Housing data—available from the UCI machine learning repository (Newman, Hettich, Blake, and Merz 1998). The data comprises observations for $n = 768$ Pima Indian women of 8 prognostic variables and the outcome (positive/negative) of a diabetes test. It is rather clear that the diabetes diagnosis depends on the plasma glucose concentration such that using a logistic regression model for diabetes explained by glucose (corresponding to $k = 2$ parameters) is intuitive. This model is partitioned with respect to the remaining $\ell = 7$ variables, using a minimal segment size of $\underline{i} = 40$ and again a Bonferroni-corrected significance level of $\alpha = 0.05$.

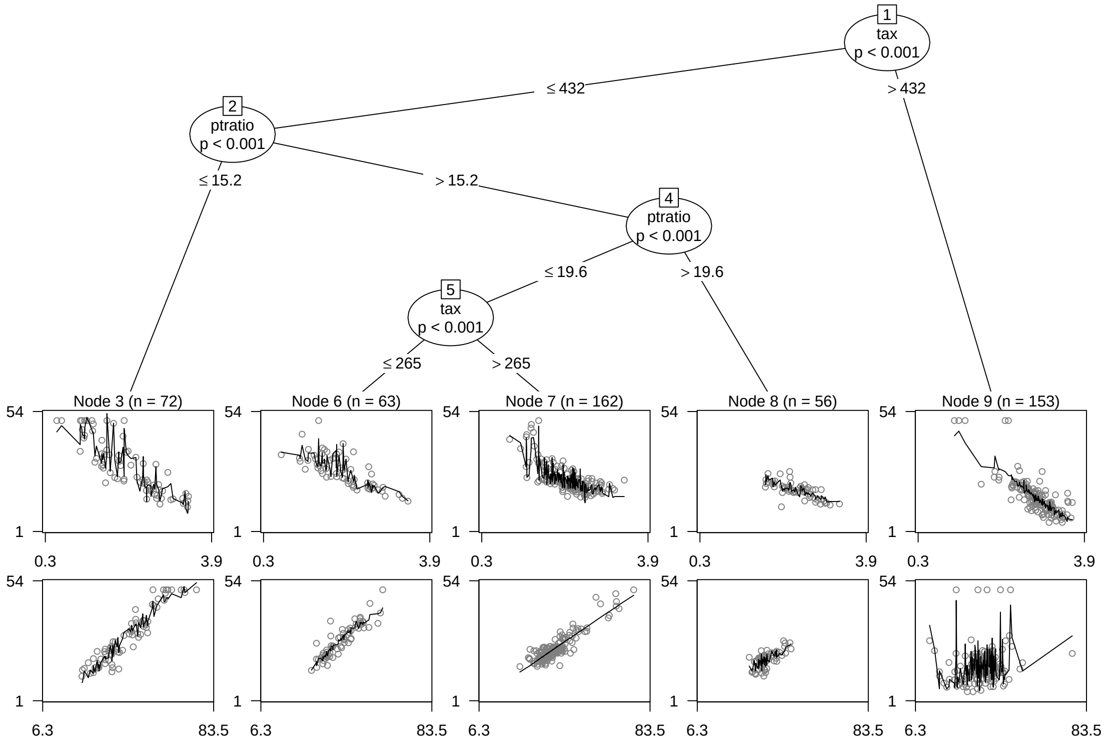

Figure 3: Linear-regression-based tree for the Boston housing data. The plots in the leaves give partial scatter plots for log(lstat) (upper panel) and $(\mathrm{rm})^2$ (lower panel).

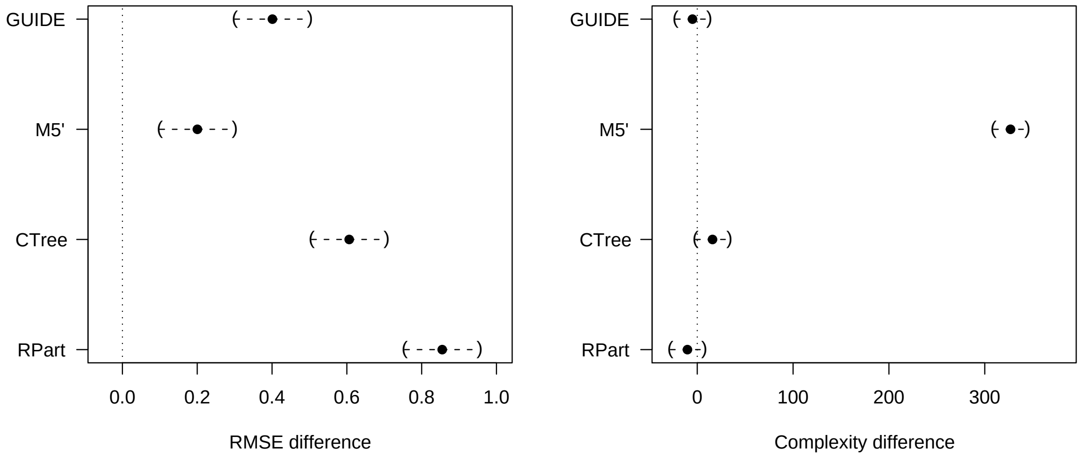

Figure 4: Performance comparison for Boston housing data: prediction error is compared by RMSE differences, complexity by difference in number of estimated parameters.

Figure 5 displays the resulting logistic regression-based tree. The data is first split at a body mass index of 26.3 (corresponding roughly to the lower quartile of this variable), those observations with a higher body mass index are partitioned into age groups below or above 30 years. The leaves of the tree visualize the data and the fitted logistic regression model using spinograms (Hofmann and Theus 2005) of diabetes by glucose (where the bins are chosen via the five point summary of glucose on the full data set). It can be seen that for women with a low body mass index the average risk of diabetes is low, but increases clearly with age (corresponding to an odds ratio of 1.060 per year). For the young women with a high body mass index, the average risk is higher and increases less quickly with respect to age (with an odds ratio of 1.048). Finally, the older women with a high body mass index have the highest average risk but with a lower odds ratio of only 1.024. The model uses 8 parameters ($3 \cdot 2$ coefficients and $3 - 1$ splits).

The results of the benchmark comparison in Table 3 and Figure 6 show that the MOB trees perform slightly (and significantly) better on this data set than the other tree-based algorithms included. In particular, it performs considerably better than the other model-based algorithm (LMT) both with respect to prediction and model complexity.

	Misclassification		Number of parameters	
	Bootstrap	Original	Bootstrap	Original
MOB	0.255	0.238	17	8
LMT	0.293	0.215	329	8
CTree	0.265	0.224	19	13
QUEST	0.265	0.250	23	3
J4.8	0.291	0.159	101	39
RPart	0.263	0.210	29	11

Table 3: Performance comparison for Pima Indians data: prediction error is compared by misclassification rate on 250 bootstrap samples and obvious misclassification on the original data set; complexity is compared by (median) number of estimated parameters.

4.4. German breast cancer study

The same ideas used for recursive partitioning in (generalized) linear regression models can straightforwardly be applied to other parametric regression models without further modification. Here, we apply the generic model-based recursive partitioning algorithm to a Weibull regression for modeling censored survival times. We follow the analysis of Schumacher, Holländer, Schwarzer, and Sauerbrei (2001) and Hothorn *et al.* (2006a) who use constant fit survival trees to analyze survival times of $n = 686$ women from positive node breast cancer in Germany. Along with the survival time (in years) and the censoring information, there are 8 covariates available as prognostic factors: number of positive lymph nodes, age, tumor size and grade, progesterone and estrogen receptor, and factors indicating menopausal status and whether the patient received a hormonal therapy.

For explaining survival from positive node breast cancer in a regression model, the number of positive lymph nodes is chosen as explanatory variable with differing intercepts depending on whether a hormonal therapy was performed or not. Together with the scale parameter of the Weibull distribution, this gives a total of $k = 4$ parameters in the model, using the remaining $\ell = 6$ prognostic variables for partitioning. The model is estimated by ML, the instability is assessed using a Bonferroni-corrected significance level of $\alpha = 0.05$ and the nodes are split with a required minimal segment size of $\underline{i} = 40$.

The resulting model-based tree is depicted in Figure 7 employing scatter plots for survival time by number of positive nodes in the leaves. Based on the ideas of Gentleman and Crowley (1991), circles with different shadings of gray (hollow and solid) are used for censored and uncensored observations, respectively. Fitted median survival times from the Weibull regression model are visualized by dashed and solid lines for patients with and with hormonal therapy. The data are

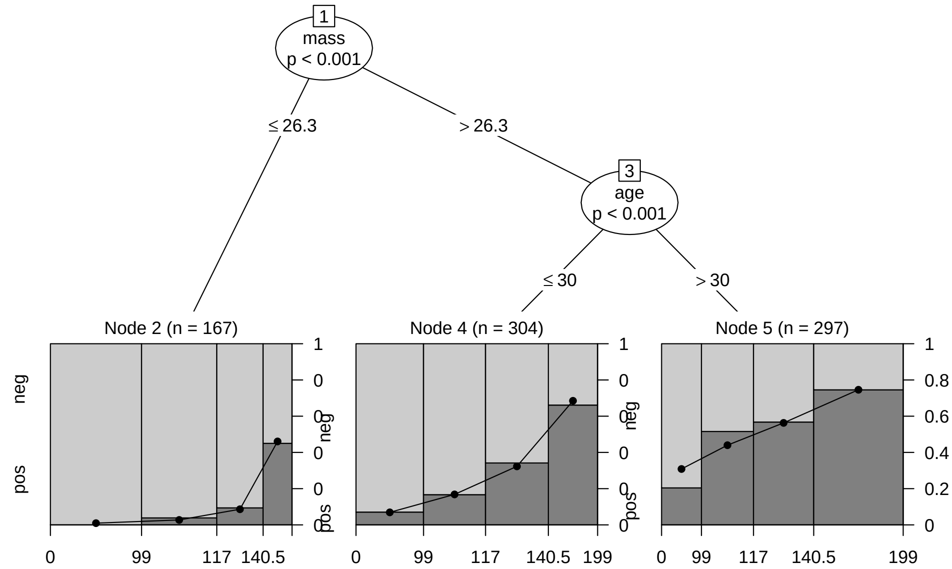

Figure 5: Logistic-regression-based tree for the Pima Indians data. The spinograms in the leaves depict diabetes by plasma glucose concentration.

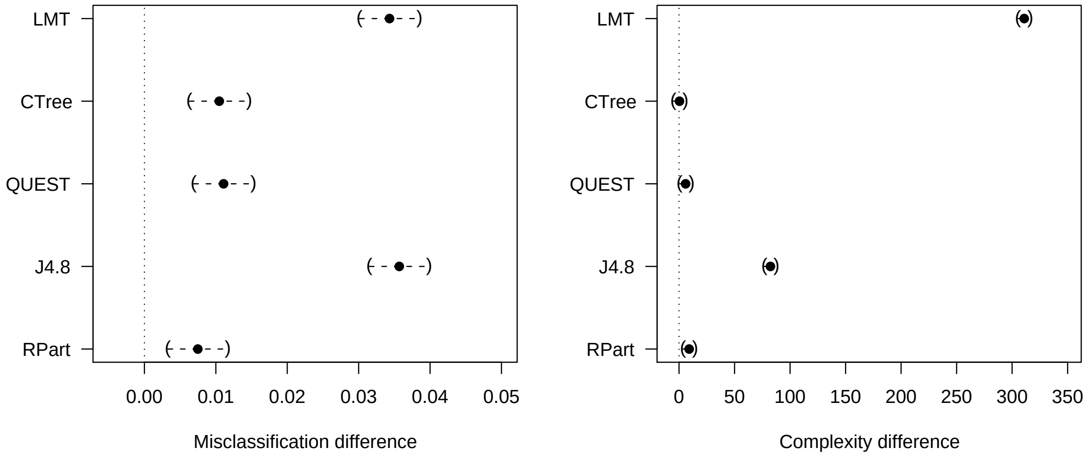

Figure 6: Performance comparison for Pima Indians data: prediction error is compared by misclassification rate differences, complexity by difference in number of estimated parameters.

partitioned once with respect to progesterone receptor splitting the observations into a group with marked influence of positive nodes and neglegible influence of hormonal therapy and a group with less pronounced influence of positive nodes but clear hormonal therapy effect. A fair amount of censored observations remains in both groups, no further significant instabilities can be detected (all p values are above 50%). The resulting model has 9 parameters ($2 \cdot 4$ coefficients and $2 - 1$ splits), yielding a log-likelihood of -809.9238.

For this survival regression problem, we refrain from conducting a benchmark comparison of performance as carried out in the previous section. The main reason for this is that it is not clear which predictive perfomance measure should be used in such a comparison: Whereas RMSE and misclassification are usually regarded to be acceptable (albeit not the only meaningful) performance measures for regression and classification tasks, the situation is not as well understood for censored regression models. Although various measures, such as the Brier score (Graf, Schmoor, Sauerbrei, and Schumacher 1999), are used in the literature their usefulness still remains a matter of debate (Henderson 1995; Altman and Royston 2000; Schemper 2003). As resolving these discussions is beyond the scope of this paper, we content ourselves with the empirical analysis for this regression problem.

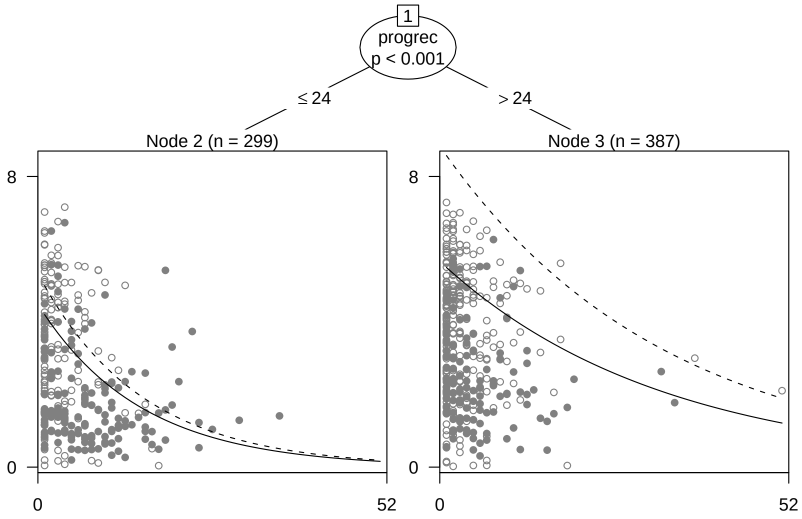

Figure 7: Weibull-regression-based tree for the German breast cancer study. The plots in the leaves depict censored (hollow) and uncensored (solid) survival time by number of positive lymph nodes along with fitted median survival for patients with (dashed line) and without (solid line) hormonal therapy.

5. Further remarks and conclusion

A powerful, flexible and unified framework for model-based recursive partitioning has been suggested. It builds on parametric models which are well-established in the statistical theory and whose parameters can be easily interpreted by subject-matter scientists. Thus, it can not only model the mean but also other properties of a parameterized distribution. Furthermore, it can be employed to partition regression relationships, such as GLMs or survival regression. It aims at minimizing a clearly defined objective function (and not certain heuristics) by a greedy forward search and is unbiased due to separation of variable and split point selection.

The algorithm as discussed in this paper relies on a statistically motivated internal stopping criterion (sometimes called *pre-pruning*), but, of course, it could also be combined with cross-validation-based *post-pruning* although the statistical interpretation of the p values would then be lost. As every node of the tree is associated with a fitted model with a certain number of parameters, another attractive option is to grow the tree with a large α and then prune based on information criteria.

For some statistical models, there is a clearly defined estimating function $\psi(Y, \theta)$ but the antiderivate $\Psi(Y, \theta)$ does not necessarily exist. Such models can also be recursively partitioned: the parameter instability tests work in the same way, only the selection of the splits has to be adapted. Instead of minimizing an objective function, the corresponding B-sample split statistics have to be maximized.

Typically, recursive partitioning algorithms use perpendicular splits, i.e., the partitioning variables Z_j just include 'main effects'. To prevent that the algorithm fails to pick up 'interaction effects' such as the XOR problem, interactions could also be added to the list of partitioning variables.

If regression models are partitioned, the question arises whether a certain covariate should be included in Y as a regressor or in Z as a partitioning variable. For categorical variables, this amounts to knowing/assuming the interactions or trying to find them adaptively—for numerical variables, it amounts to knowing/assuming a segment-wise linear relationship vs. approximating a possibly non-linear influence by a step function. The separation can usually be made based on subject knowledge as in the economic journals example. Other scenarios of this kind are conceivable: e.g., in biostatistics it would be natural to fit a dose-response relationship and partition it with respect to further experiment-specific covariables, or in business applications a market segmentation could be carried out based on a standard demand function. Finally, the variables entering the explanatory part of Y and Z can also be overlapping, however, a trend-resistant fluctuation test should be conducted for partitioning.

Within the genuine statistical framework proposed in this paper, practitioners can assess whether one (standard) global parametric model fits their data or whether it is more appropriate to partition it with respect to further covariates. If so, the partitioning variables and their split points are selected separately in a forward search that controls the type I error rates for the variable selection in each node. This formulation of the algorithm ensures that interpretations obtained from graphical representations of the corresponding tree-structured models are valid in a statistical sense.

References

Altman DG, Royston P (2000). "What Do We Mean by Validating a Prognostic Model?" *Statistics in Medicine*, **19**, 453–473.

Andrews DWK (1993). "Tests for Parameter Instability and Structural Change With Unknown Change Point." *Econometrica*, **61**, 821–856.

Andrews DWK, Ploberger W (1994). "Optimal Tests When a Nuisance Parameter is Present Only Under the Alternative." *Econometrica*, **62**, 1383–1414.

Bai J, Perron P (2003). "Computation and Analysis of Multiple Structural Change Models." *Journal of Applied Econometrics*, **18**, 1–22.

Bergstrom TC (2001). "Free Labor for Costly Journals?" *Journal of Economic Perspectives*, **15**, 183–198.

Breiman L (2001). "Random Forests." *Machine Learning*, **45**(1), 5–32.

Breiman L, Friedman JH (1985). "Estimating Optimal Transformations for Multiple Regression and Correlation." *Journal of the American Statistical Association*, **80**(391), 580–598.

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984). *Classification and Regression Trees.* Wadsworth, California.

Bühlmann P, Yu B (2003). "Boosting with L_2 Loss: Regression and Classification." *Journal of the American Statistical Association*, **98**(462), 324–338.

Chan KY, Loh WY (2004). "LOTUS: An Algorithm for Building Accurate and Comprehensible Logistic Regression Trees." *Journal of Computational and Graphical Statistics*, **13**(4), 826–852.

Choi Y, Ahn H, Chen JJ (2005). "Regression Trees for Analysis of Count Data With Extra Poisson Variation." *Computational Statistics & Data Analysis*, **49**, 893–915.

Chow GC (1960). "Tests of Equality Between Sets of Coefficients in Two Linear Regressions." *Econometrica*, **28**, 591–605.

Chu CSJ, Hornik K, Kuan CM (1995). "MOSUM Tests for Parameter Constancy." *Biometrika*, **82**, 603–617.

Gama J (2004). "Functional Trees." *Machine Learning*, **55**, 219–250.

Gentleman R, Crowley J (1991). "Graphical Methods for Censored Data." *Journal of the American Statistical Association*, **86**, 678–683.

Graf E, Schmoor C, Sauerbrei W, Schumacher M (1999). "Assessment and comparison of prognostic classification schemes for survival data." *Statistics in Medicine*, **18**(17-18), 2529–2545.

Hansen BE (1997). "Approximate Asymptotic p Values for Structural-Change Tests." *Journal of Business & Economic Statistics*, **15**, 60–67.

Hawkins DM (2001). "Fitting Multiple Change-Point Models to Data." *Computational Statistics & Data Analysis*, **37**, 323–341.

Henderson R (1995). "Problems and Prediction in Survival-Data Analysis." *Statistics in Medicine*, **14**, 161–184.

Hjort NL, Koning A (2002). "Tests for Constancy of Model Parameters Over Time." *Nonparametric Statistics*, **14**, 113–132.

Hofmann H, Theus M (2005). "Interactive Graphics for Visualizing Conditional Distributions." Unpublished Manuscript.

Hornik K, Zeileis A, Hothorn T, Buchta C (2006). **RWeka***: An R Interface to* **Weka**. R package version 0.2-2.

Hothorn T, Hornik K, Zeileis A (2006a). "Unbiased Recursive Partitioning: A Conditional Inference Framework." *Journal of Computational and Graphical Statistics*. Forthcoming, URL http://statmath.wu-wien.ac.at/~zeileis/papers/Hothorn+Hornik+Zeileis-2006.pdf.

Hothorn T, Leisch F, Zeileis A, Hornik K (2005). "The Design and Analysis of Benchmark Experiments." *Journal of Computational and Graphical Statistics*, **14**(3), 675–699.

Hothorn T, Zeileis A, Hornik K (2006b). **party***: A Laboratory for Recursive Part(y)itioning*. R package version 0.8-3.

Kim H, Loh WY (2001). "Classification Trees With Unbiased Multiway Splits." *Journal of the American Statistical Association*, **96**(454), 589–604.

Landwehr N, Hall M, Frank E (2005). "Logistic Model Trees." *Machine Learning*, **59**, 161–205.

Loh WY (2002). "Regression Trees With Unbiased Variable Selection and Interaction Detection." *Statistica Sinica*, **12**, 361–386.

Loh WY, Shih YS (1997). "Split Selection Methods for Classification Trees." *Statistica Sinica*, **7**, 815–840.

Meyer D, Leisch F, Hornik K (2003). "The Support Vector Machine Under Test." *Neurocomputing*, **55**(1–2), 169–186.

Morgan JN, Sonquist JA (1963). "Problems in the Analysis of Survey Data, and a Proposal." *Journal of the American Statistical Association*, **58**, 415–434.

Muggeo VMR (2003). "Estimating Regression Models With Unknown Break-Points." *Statistics in Medicine*, **22**, 3055–3071.

Newman DJ, Hettich S, Blake CL, Merz C (1998). "UCI Repository of Machine Learning Databases." URL http://www.ics.uci.edu/~mlearn/MLRepository.html.

Nyblom J (1989). "Testing for the Constancy of Parameters Over Time." *Journal of the American Statistical Association*, **84**, 223–230.

O'Brien SM (2004). "Cutpoint Selection for Categorizing a Continuous Predictor." *Biometrics*, **60**, 504–509.

Ploberger W, Krämer W (1992). "The CUSUM Test With OLS Residuals." *Econometrica*, **60**(2), 271–285.

Potts D, Sammut C (2005). "Incremental Learning of Linear Model Trees." *Machine Learning*, **61**, 5–48.

Quinlan JR (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publ., San Mateo, California.

Quinlan R (1992). "Learning with Continuous Classes." In "Proceedings of the Australian Joint Conference on Artificial Intelligence," pp. 343–348. World Scientific, Singapore.

R Development Core Team (2005). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-00-3, URL http://www.R-project.org/.

Samarov A, Spokoiny V, Vial C (2005). "Component Indentification and Estimation in Nonlinear High-Dimension Regression Models by Structural Adaptation." *Journal of the American Statistical Association*, **100**(470), 429–445.

Schemper M (2003). "Predictive Accuracy and Explained Variation." *Statistics in Medicine*, **22**, 2299–2308.

Schumacher M, Holländer N, Schwarzer G, Sauerbrei W (2001). "Prognostic Factor Studies." In J Crowley (ed.), "Statistics in Clinical Oncology," pp. 321–378. Marcel Dekker, New York.

Stock JH, Watson MW (2003). *Introduction to Econometrics*. Addison Wesley.

Su X, Wang M, Fan J (2004). "Maximum Likelihood Regression Trees." *Journal of Computational and Graphical Statistics*, **13**, 586–598.

Therneau TM, Atkinson EJ (1997). "An Introduction to Recursive Partitioning Using the **rpart** Routine." *Technical Report 61*, Section of Biostatistics, Mayo Clinic, Rochester. URL http://www.mayo.edu/hsr/techrpt/61.pdf.

Vapnik VN (1996). *The Nature of Statistical Learning Theory*. Springer, New York.

Wang Y, Witten IH (1997). "Induction of Model Trees for Predicting Continuous Classes." In "Proceedings of the European Conference on Machine Learning," University of Economics, Faculty of Informatics and Statistics, Prague.

White H (1994). *Estimation, Inference and Specification Analysis*. Cambridge University Press.

Witten IH, Frank E (2005). *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, San Francisco, 2nd edition.

Zeileis A (2005). "A Unified Approach to Structural Change Tests Based on ML Scores, *F* Statistics, and OLS Residuals." *Econometric Reviews*, **24**, 445–466.

Zeileis A, Hornik K (2003). "Generalized M-Fluctuation Tests for Parameter Instability." *Report 80*, SFB "Adaptive Information Systems and Modelling in Economics and Management Science". URL http://www.wu-wien.ac.at/am/reports.htm#80.

Zeileis A, Kleiber C, Krämer W, Hornik K (2003). "Testing and Dating of Structural Changes in Practice." *Computational Statistics & Data Analysis*, **44**(1–2), 109–123.

A. R code

The following R code is sufficient for reproducing the empirical examples from Section 4 including data pre-processing, model fitting and visualization.

```
library("party")

data("Journals", package = "Ecdat")
journals <- Journals[, c("libprice", "society", "citestot")]
journals$oclc <- log(Journals$oclc)
journals$citeprice <- log(Journals$libprice/Journals$citestot)
journals$age <- 2000 - Journals$date1
journals$chars <- Journals$charpp*Journals$pages/10^6
mobJ <- mob(oclc ~ citeprice | society + citestot + age + chars + libprice,
  data = journals, model = linearModel, control = mob_control(minsplit = 10))
plot(mobJ)

data("BostonHousing", package = "mlbench")
BostonHousing$lstat <- log(BostonHousing$lstat)
BostonHousing$rm <- BostonHousing$rm^2
BostonHousing$chas <- factor(BostonHousing$chas)
BostonHousing$rad <- factor(BostonHousing$rad, ordered = TRUE)
mobBH <- mob(medv ~ lstat + rm | zn + indus + chas + nox + age + dis + rad +
  tax + crim + b + ptratio, data = BostonHousing,
  control = mob_control(minsplit = 40), model = linearModel)
plot(mobBH)

data("PimaIndiansDiabetes", package = "mlbench")
mobPID <- mob(diabetes ~ glucose | pregnant + pressure + triceps + insulin +
  mass + pedigree + age, data = PimaIndiansDiabetes, model = glinearModel,
  control = mob_control(minsplit = 40), family = binomial())
plot(mobPID)

data("GBSG2", package = "ipred")
nloglik <- function(x) -logLik(x)
GBSG2$time <- GBSG2$time/365
mobGBSG2 <- mob(Surv(time, cens) ~ horTh + pnodes | progrec + menostat +
  estrec + menostat + age + tsize + tgrade, data = GBSG2, model = survReg,
  control = mob_control(objfun = nloglik, minsplit = 40))
plot(mobGBSG2, terminal = node_scatterplot, tp_args = list(yscale = c(-0.1, 11)))
```