WIRTSCHAFTS UNIVERSITÄT WIEN

Unbiased Recursive Partitioning II:

A Parametric Framework Based on Parameter Instability Tests

Achim Zeileis Torsten Hothorn Kurt Hornik

http://statmath.wu-wien.ac.at/~zeileis/

Overview

- Model-based recursive partitioning
  - Parametric models
  - Parameter estimation
  - Segmented models
- The recursive partitioning algorithm
  - Tests for parameter instability
  - Assessing numerical/categorical variables
- Illustrations
  - Pima Indians diabetes data
  - Boston housing data
- Summary

Model-based recursive partitioning

Starting point: Recursive partitioning algorithms (including conditional inference trees) learn a partition/segmentation from data and then fit a naive model in each terminal node, e.g., a mean, relative frequencies or a Kaplan-Meier curve.

Idea: Employ parametric models in each node.

Goal: Algorithm for constructing segmented parametric models by recursive partitioning.

Parametric models

Consider models $\mathcal{M}(Y, \theta)$ 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 fit by minimizing some objective function $\Psi(Y, \theta)$ yielding the parameter estimate $\widehat{\theta}$

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

Parameter estimation

Under mild regularity conditions it can be shown that the estimate $\widehat{\theta}$ can also be computed by solving the first order conditions

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

where

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

is the score function or estimating function corresponding to $\Psi(Y, \theta)$.

Parameter estimation

This type of estimators includes maximum likelihood (ML), ordinary least squares (OLS), Quasi-ML and further M-type estimators.

Example: $\mathcal{M}(Y, \theta)$ could be a multivariate normal model for $Y \sim \mathcal{N}(\mu, \Sigma)$ such that $\theta = (\mu, \Sigma)$.

Example: $\mathcal{M}(Y, \theta)$ could be a generalized linear model for $Y = (y, x)^\top$ such that

$$g(\mathsf{E}(y)) \quad = \quad x^\top \theta.$$

Segmented models

Idea: In many situations, it is unreasonable to assume that a single global model $\mathcal{M}(Y, \theta)$ can be fit to **all** n observations. But it might be possible to partition the observations with respect to covariates $Z = (Z_1, \dots, Z_l)$ such that a fitting model can be found in each cell of the partition.

Goal: Learn partition via recursive partitioning with respect to $Z_j \in \mathcal{Z}_j$ $(j = 1, \dots, l)$.

Segmented models

Example: Regression trees.
The parameter θ describes the mean of the univariate observations Y_i and is estimated by OLS or equivalently ML in a normal model. The variables Z_j are the regressors considered for partitioning.

Example: Changepoint or structural change analysis.
A (generalized) linear regression model with $Y_i = (y_i, x_i)^\top$ and regression coefficients θ is segmented with respect to a single variable Z_1 (i.e., $l = 1$), typically time.

Segmented models

Given a partition, 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 $\widehat{\theta}^{(b)}$ in each segment b (with corresponding indices I_b).

If it is unknown, minimization of Ψ is more complicated (if trivial partitions are excluded). But it is easily possible to optimally split the observations with respect to only a single variable Z_1 into B segments. Typically $B = 2$ is chosen.

Segmented models

A single optimal split into $B = 2$ partitions can easily be computed in $O(n)$ by exhaustive search.

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 $O(n^2)$ (Hawkins, 2001; Bai & Perron, 2003) or via iterative algorithms (Muggeo, 2003).

Various algorithms for adaptively choosing the number of segments B are available, e.g., via information criteria.

The recursive partitioning algorithm

The generic recursive partitioning algorithm presented in Part I can be used almost directly.

The only difference is that now each node is associated with a parametric model.

Question: How should we assess the association of a fitted model with a covariate Z_j?

Answer: Test for instability of the parameters of the model with respect to this variable Z_j.

The recursive partitioning algorithm

1. Fit the model once to all observations in the current node by estimating $\widehat{\theta}$ via minimization of Ψ.
2. Assess whether the parameter estimates are stable with respect to every ordering $Z_1, \dots, Z_l$. 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 number of splits, or choose the number of splits adaptively).
4. Split this node into daughter nodes and repeat the procedure.

Tests for parameter instability

Generalized M-fluctuation tests (Zeileis & Hornik, 2003) can be used for assessing whether the parameter estimates $\widehat{\theta}$ are stable over a certain variable or not.

The basic idea is to use an empirical fluctuation process of cumulative scores for a particular ordering of the observations

$$W(t, \widehat{\theta}) \quad = \quad \widehat{J}^{-1/2} n^{-1/2} \sum_{i=1}^{\lfloor nt \rfloor} \psi(Y_i, \widehat{\theta}) \qquad (0 \le t \le 1)$$

which is governed by a functional central limit theorem (FCLT). It converges to a Brownian bridge W^0.

Tests for parameter instability

A test statistic can be derived by applying a scalar functional $\lambda(\cdot)$ to the fluctuation process, the limiting distribution is just the same functional (or its asymptotical counterpart) applied to the limiting process $\lambda(W^0(\cdot))$.

Advantage: The model just has to be estimated once. For testing, the scores of the fitted model $\widehat{\psi}$ just have to be re-ordered for each variable.

Let $W_j(t)$ be the fluctuation process for the observations ordered by Z_j.

Assessing numerical variables

The most intuitive functional for assessing the stability with respect to a numerical partitioning variable Z_j is the supLM statistic of Andrews (1993).

$$\lambda_{\mathrm{supLM}}(W_j) \quad = \quad \max_{i=\underline{i},\dots,\bar{\imath}} \left(\frac{i}{n} \cdot \frac{n-i}{n}\right)^{-1} \left\| W_j\left(\frac{i}{n}\right) \right\|_2^2 .$$

This gives the maximum of the single changepoint LM statistics over all possible changepoints in $[\underline{i}, \bar{\imath}]$.

The limiting distribution is given by the supremum of a squared, k-dimensional tied-down Bessel process.

Assessing categorical variables

To assess the stability of a categorical variable with C levels, a χ^2 statistic is most intuitive

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

because it is insensitive to re-ordering of the levels and the observations within the levels.

It essentially captures the instability when splitting the model into C groups.

The limiting distribution is χ^2 with $k \cdot (C - 1)$ degrees of freedom.

Pruning

The algorithm described so far employs a **pre-pruning** strategy, i.e., uses an internal stopping criterion: if no variable exhibits significant association, i.e., significant parameter instability, the algorithm stops.

Alternatively/additionally, a **post-pruning** strategy can be used. This seems particularly attractive if ML is used for parameter estimation. Then a ML tree can be grown which is consequently associated with a segmented ML model. This can be pruned afterwards using information criteria for example.

Example: Pima Indians diabetes data

Goal: Explain outcome of a test for diabetes among Pima Indian women.

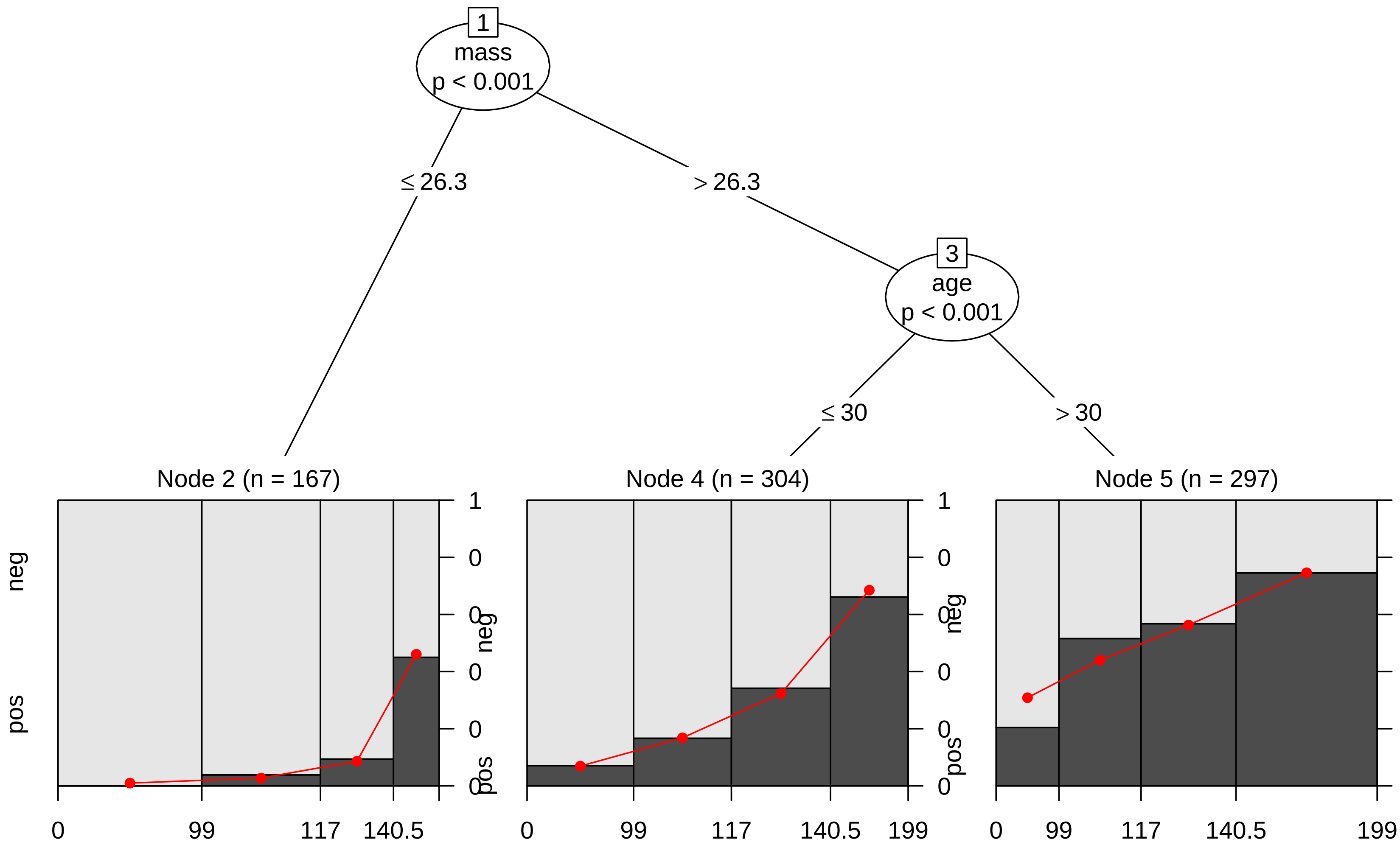
Clear: Outcome depends on plasma glucose concentration.

Here: Segment a logistic regression with explanatory variable glucose. All remaining variables are used as partitioning variables.

Example: Pima Indians diabetes data

```
R> fmPID <- mob(diabetes ~ glucose | pregnant + pressure + triceps +
+    insulin + mass + pedigree + age, data = PimaIndiansDiabetes,
+    model = glinearModel, family = binomial())

-------------------------------------------
Fluctuation tests of splitting variables:
              pregnant  pressure    triceps   insulin         mass
statistic 2.988568e+01 7.5028009 15.94104656 6.5968798 4.881000e+01
p.value   1.080668e-04 0.9745212  0.09443482 0.9951636 5.006389e-09
             pedigree          age
statistic 18.33529248 4.351454e+01
p.value    0.03262343 8.545400e-08

Best splitting variable: mass
Perform split? yes
-------------------------------------------
Node properties:
mass <= 26.3; criterion = 1, statistic = 48.81

...

R> plot(fmPID)
```

Example: Pima Indians diabetes data



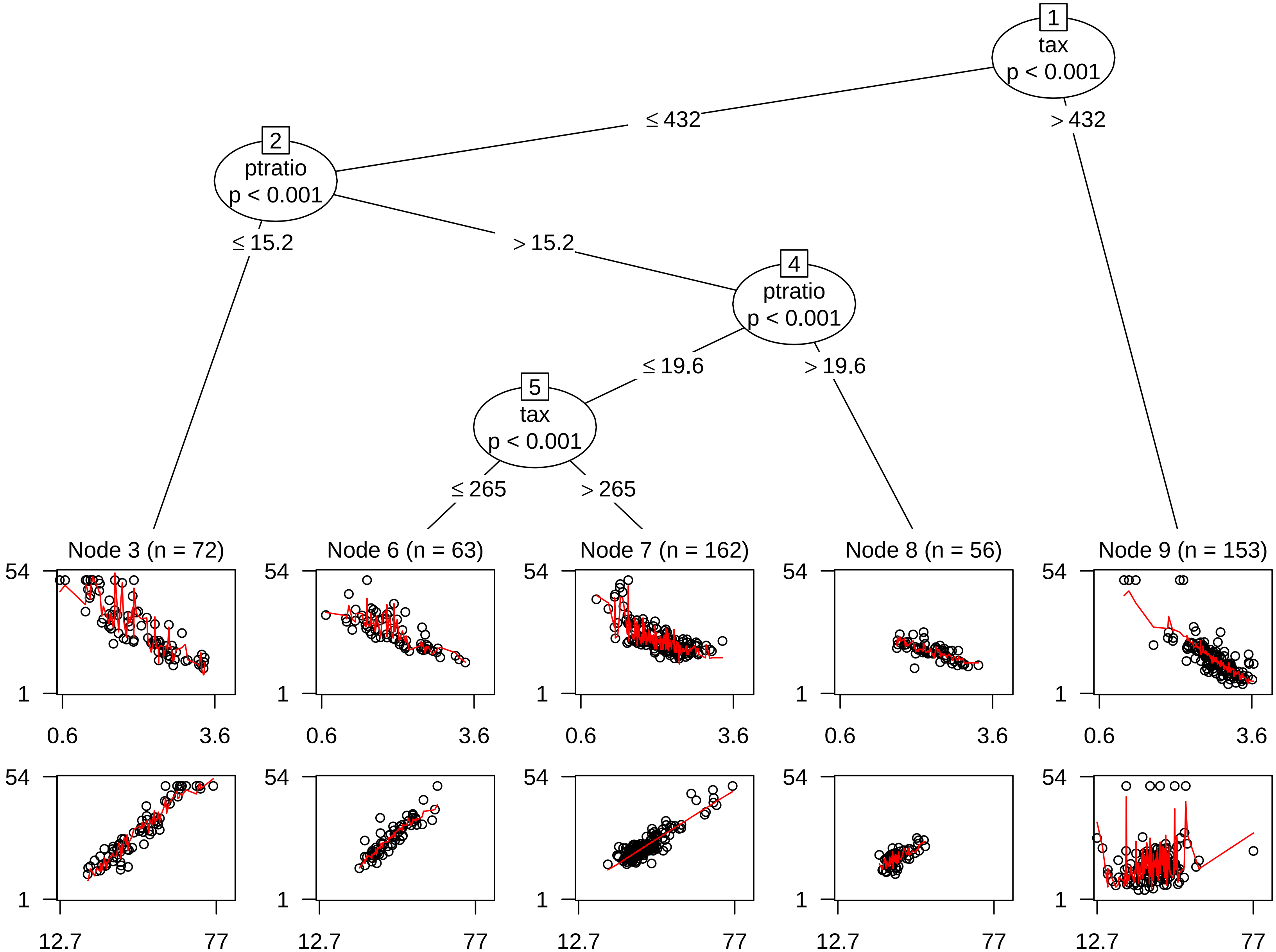
Example: Boston housing data

Goal: Explain median value of houses in suburbs of Boston by various numerical covariates.

Clear: House value depends on number of rooms and lower status percentage.

Here: Segment a linear regression with explanatory variables (average number of rooms)^2 and log(lower status percentage). All remaining variables are used as partitioning variables.

Example: Boston housing data



Summary

Model-based recursive partitioning:

- based on well-established statistical models,
- aims at minimizing a clearly defined objective function (and not certain heuristics),
- unbiased due to separation of variable and cutpoint selection,
- statistically motivated stopping criterion,
- employs general class of tests for parameter instability.
- available in function `mob()` in package **party** available from

  http://CRAN.R-project.org/