University of Innsbruck

partykit: A modular toolkit for recursive partytioning in R

Torsten Hothorn, Achim Zeileis

Working Papers in Economics and Statistics

2014-10

Research Platform
eeecon
Empirical and Experimental Economics
University of Innsbruck
http://eeecon.uibk.ac.at/

University of Innsbruck
Working Papers in Economics and Statistics

The series is jointly edited and published by

- Department of Economics
- Department of Public Finance
- Department of Statistics

Contact Address:
University of Innsbruck
Department of Public Finance
Universitaetsstrasse 15
A-6020 Innsbruck
Austria
Tel: + 43 512 507 7171
Fax: + 43 512 507 2970
E-mail: eeecon@uibk.ac.at

The most recent version of all working papers can be downloaded at
http://eeecon.uibk.ac.at/wopec/

For a list of recent papers see the backpages of this paper.

partykit: A Modular Toolkit for Recursive Partytioning in R

Torsten Hothorn
Universität Zürich

Achim Zeileis
Universität Innsbruck

Abstract

The R package **partykit** provides a flexible toolkit for learning, representing, summarizing, and visualizing a wide range of tree-structured regression and classification models. The functionality encompasses: (a) basic infrastructure for *representing* trees (inferred by any algorithm) so that unified `print`/`plot`/`predict` methods are available; (b) dedicated methods for trees with *constant fits* in the leaves (or terminal nodes) along with suitable coercion functions to create such trees (e.g., by **rpart**, **RWeka**, PMML); (c) a reimplementation of *conditional inference trees* (`ctree`, originally provided in the **party** package); (d) an extended reimplementation of *model-based recursive partitioning* (`mob`, also originally in **party**) along with dedicated methods for trees with parametric models in the leaves. Here, a brief overview of the package and its design is given while more detailed discussions of items (a)–(d) are available in vignettes accompanying the package.

Keywords: recursive partitioning, regression trees, classification trees, statistical learning, R.

1. Overview

In the more than fifty years since Morgan and Sonquist (1963) published their seminal paper on "automatic interaction detection", a wide range of methods has been suggested that is usually termed "recursive partitioning" or "decision trees" or "tree(-structured) models" etc. The particularly influential algorithms include CART (classification and regression trees, Breiman, Friedman, Olshen, and Stone 1984), C4.5 (Quinlan 1993), QUEST/GUIDE (Loh and Shih 1997; Loh 2002), and CTree (Hothorn, Hornik, and Zeileis 2006) among many others (see Loh 2014, for a recent overview). Reflecting the heterogeneity of conceptual algorithms, a wide range of computational implementations in various software systems emerged: Typically the original authors of an algorithm also provide accompanying software but many software systems, including **Weka** (Witten and Frank 2005) or R (R Core Team 2013), also provide collections of various types of trees. Within R the list of prominent packages includes **rpart** (Therneau and Atkinson 1997, implementing CART), **RWeka** (Hornik, Buchta, and Zeileis 2009, with interfaces to J4.8, M5', LMT from **Weka**), and **party** (Hothorn, Hornik, Strobl, and Zeileis 2013, implementing CTree and MOB) among many others. See the CRAN task view "Machine Learning" (Hothorn 2014) for an overview.

All of these algorithms and software implementations have to deal with similar challenges. However, due to the fragmentation of the communities in which they are published – ranging from statistics over machine learning to various applied fields – many discussions of the algorithms do not reuse established theoretical results and terminology. Similarly, there is

no common "language" for the software implementations and different solutions are provided by different packages (even within R) with relatively little reuse of code. The **partykit** aims at mitigating the latter issue by providing a common unified infrastructure for recursive partytioning in the R system for statistical computing. In particular, **partykit** provides tools for representing, printing, plotting trees and computing predictions. The design principles are:

- One 'agnostic' base class ('`party`') encompassing a very wide range of different tree types.
- Subclasses for important types of trees, e.g., trees with constant fits ('`constparty`') or with parametric models ('`modelparty`') in each terminal node (or leaf).
- Nodes are recursive objects, i.e., a node can contain child nodes.
- Keep the (learning) data out of the recursive node and split structure.
- Basic printing, plotting, and predicting for raw node structure.
- Customization via suitable panel or panel-generating functions.
- Coercion from existing object classes in R (`rpart`, `J48`, etc.) to the new class.
- Usage of simple/fast S3 classes and methods.

In addition to all of this generic infrastructure, two specific tree algorithms are implemented in **partykit** as well: `ctree` for conditional inference trees (Hothorn *et al.* 2006) and `mob` for model-based recursive partitioning (Zeileis, Hothorn, and Hornik 2008).

2. Installation and Documentation

The **partykit** package is an add-on package for the R system for statistical computing. It is available from the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=partykit` and can be installed from within R, e.g., using `install.packages`. It depends on R (at least 2.15.0) as well as the base packages **graphics**, **grid**, **stats**, and the recommended **survival**. Furthermore, various suggested packages are needed for certain special functionalities in the package. To install all of these required and suggested packages in one go, the command `install.packages("partykit", dependencies = TRUE)` can be used.

In addition to the stable release version on CRAN, the current development release can be installed from R-Forge (Theußl and Zeileis 2009). In addition to source and binary packages the entire version history is available through R-Forge's **Subversion** source code management system.

Along with the package extensive documentation with examples is shipped. The manual pages provide basic technical information on all functions while much more detailed descriptions along with hands-on examples are provided in the four package vignettes. First, the vignette `"partykit"` introduces the basic '`party`' class and associated infrastructure while three further vignettes discuss the tools built on top of it: `"constparty"` covers the eponymous

class (as well as the simplified '`simpleparty`' class) for constant-fit trees along with suitable coercion functions, and `"ctree"` and `"mob"` discuss the new `ctree` and `mob` implementations, respectively. Each of the vignettes can be viewed within R via `vignette("name", package = "partykit")` and the underlying source code (in R with LaTeX text) is also available in the source package.

3. User Interface

The **partykit** package provides functionality at different levels. First, there is basic infrastructure for representing, modifying, and displaying trees and recursive partitions – these tools are mostly intended for developers and described in the next section. Second, there are tools for inferring trees from data or for importing trees inferred by other software into **partykit**.

While originally an important goal for the development of **partykit** was to provide infrastructure for the authors' own tree induction algorithms CTree and MOB, the design was very careful to separate as much functionality as possible into more general classes that are useful for a far broader class of trees. In particular, to be able to print/plot/predict different trees in a unified way, there are so-called coercion functions for transforming trees learned in other software packages (inside and outside of R) to the classes provided by **partykit**. Specifically, tree objects learned by `rpart` (Therneau and Atkinson 1997, implementing CART, Breiman *et al.* 1984) and by `J48` from **RWeka** (Hornik *et al.* 2009, interfacing **Weka**'s J4.8 algorithm for C4.5, Quinlan 1993) can be coerced by `as.party` to the same object class '`constparty`'. This is a general class that can in principle represent all the major classical tree algorithms with constant fits in the terminal nodes. Also, the same class is employed for conditional inference trees (CTree) that can be learned with the `ctree` function directly within **partykit** or evolutionary trees from package **evtree** (Grubinger, Zeileis, and Pfeiffer 2011).

Not only trees learned within R can be transformed to the proposed infrastructure but also trees from other software packages. Either a dedicated interface has to be created using the building blocks described in the next section (e.g., as done for the J4.8 tree in **RWeka**) or PMML (Predictive Model Markup Language) can be used as an intermediate exchange format. This is an XML standard created by an international consortium (Data Mining Group 2014) that includes a `<TreeModel>` tag with support for constant-fit classification and regression trees. The function `pmmlTreeModel` allows to read these files and represents them as '`simpleparty`' objects in **partykit**. The reason for not using the '`constparty`' class as above is that the PMML format only stores point predictions (e.g., a mean or proportion)

Algorithm	Software	Object class	Original reference
CART/RPart	`rpart::rpart` + `as.party`	`constparty`	Breiman *et al.* (1984)
C4.5/J4.8	**Weka**/`RWeka::J48` + `as.party`	`constparty`	Quinlan (1993)
QUEST	SPSS/**AnswerTree** + `pmmlTreeModel`	`simpleparty`	Loh and Shih (1997)
CTree	`ctree`	`constparty`	Hothorn *et al.* (2006)
MOB	`mob`, `lmtree`, `glmtree`, ...	`modelparty`	Zeileis *et al.* (2008)
EvTree	`evtree::evtree`	`constparty`	Grubinger *et al.* (2011)

Table 1: Selected tree algorithms than can be interfaced through **partykit**. The software column lists external software, R functions from other packages (with `::` syntax) and from **partykit**.

rather than all observations from the learning sample. So far, the PMML interface has been tested with output from the R package **pmml** and SPSS's **AnswerTree** model. The latter includes an implementation of the QUEST algorithm (Loh and Shih 1997).

Finally, the **partykit** function `mob` implements model-based recursive partitioning (MOB) along with "mobster" interfaces for certain models (e.g., `lmtree`, `glmtree`). These return objects of class '`modelparty`' where nodes are associated with statistical models (as opposed to simple constant fits). In principle, this may also be adapted to other model trees (such as GUIDE, LMT, or M5') but no such interface is currently available.

All of these different trees (see Table 1 for an overview) use the same infrastructure at the core but possibly with different options enabled. In all cases, the functions `print`, `plot`, and `predict` can be used to create textual and graphical displays of the tree and for computing predictions on new data, respectively. As an example for the visualizations, Figure 1 shows two different trees fitted to the well-known data on survival of passengers on the ill-fated maiden voyage of the RMS Titanic: The upper panel shows a CART tree with constant fits learned by **rpart** and converted to **partykit**. The lower panel shows a MOB tree learned with **partykit** with a logistic regression for treatment effects in the terminal nodes. Additionally, the are further utility functions, e.g., `nodeapply` can be employed to access further information stored in the nodes of a tree and `nodeprune` can prune selected nodes.

4. Developer Infrastructure

The unified infrastructure at the core of **partykit** is especially appealing for developers who either want to implement new tree algorithms or represent trees learned in other systems.

Here, we briefly outline the most important classes and refer to the vignettes for more details:

'`partysplit`': Split specification with an integer ID for the splitting variable, breakpoint(s), and indexes for the kids.

'`partynode`': Node specification with an integer ID, a '`partysplit`', and a list of kids (if any) that are '`partynode`' objects again.

'`party`': Tree specification with a '`partynode`' tree along with a '`data.frame`' (which optionally may be empty), potentially plus further information about fitted values and so-called '`terms`' that allow to preprocess new data for predictions.

All classes have an additional slot for storing arbitrary information at any level of the tree. This is exploited by '`constparty`', '`simpleparty`', and '`modelparty`' which store the observed response, point predictions, and fitted parametrics models, respectively.

5. Discussion and Outlook

Package **partykit** provides a toolkit for trees in R that gives emphasis to flexibility and extensibility. The infrastructure is easily accessible and written in plain R. Nevertheless speed is not a critical concern because (a) lean data structures are employed and (b) the time-intensive part in tree computations is typically inferring the tree rather than representing/printing/plotting it. Hence, the two functions for inferring trees either have computationally intensive parts in

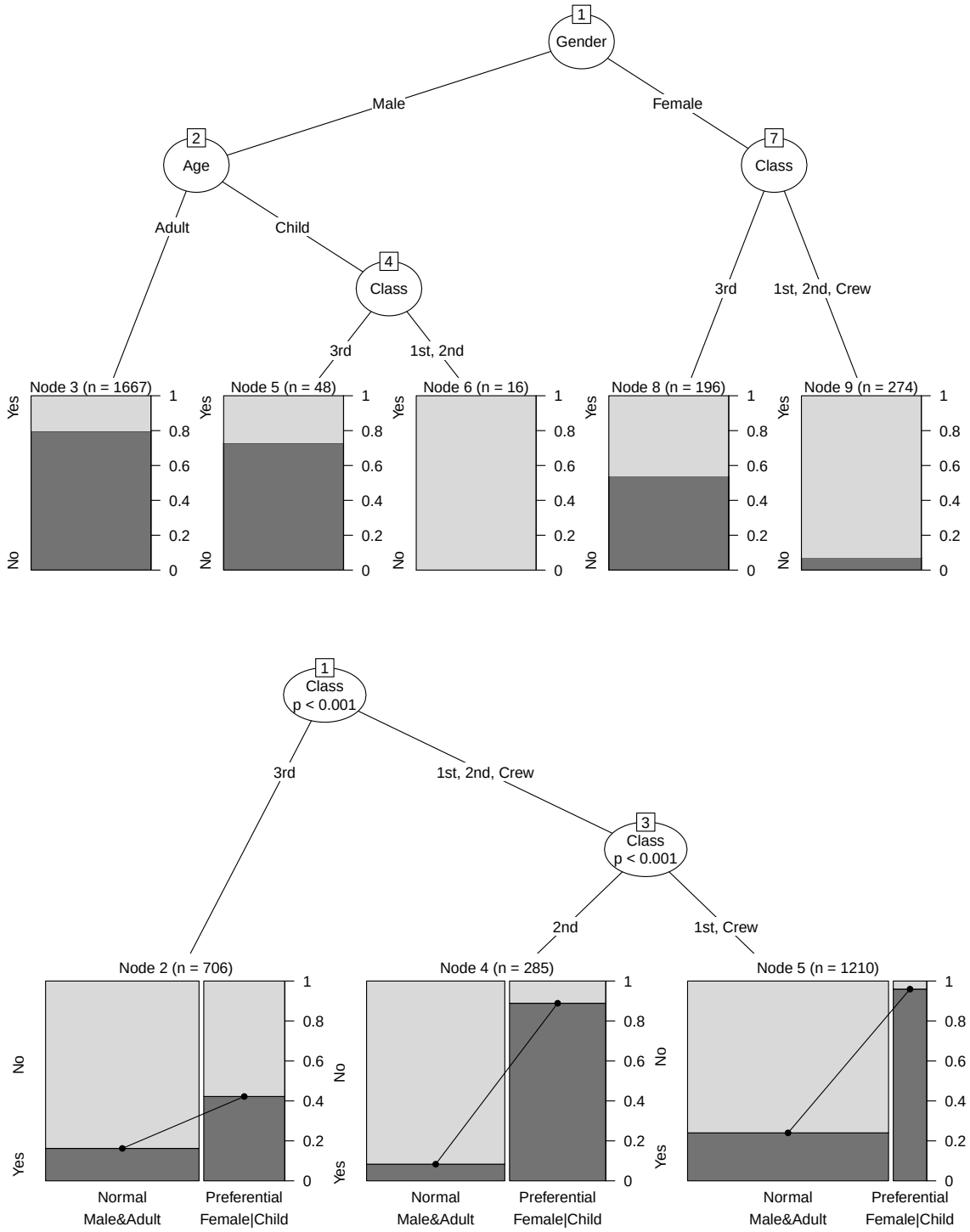

Figure 1: Tree visualizations of survival on Titanic: '`rpart`' tree converted with `as.party` and visualized by **partykit** (top); and logistic-regression-based tree fitted by `glmtree` (bottom).

C (`ctree`) or build on R's fitting functions (`mob`). For benchmarks of their predictive performance, see their original references. Regarding memory requirements, the '`party`' trees are very lean but can optionally be used to store any information. The recursive node structure itself is extremely lightweight and optionally zero rows from the learning data are sufficient, which future versions will exploit for storing forests.

Acknowledgments

We are thankful to the organizers and participants of the "Workshop on Classification and Regression Trees" (March 2014), sponsored by the Institute for Mathematical Sciences of the National University of Singapore, for helpful feedback and stimulating discussions.

References

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

Data Mining Group (2014). "Predictive Model Markup Language." Version 4.2, URL http://www.dmg.org/.

Grubinger T, Zeileis A, Pfeiffer KP (2011). "**evtree**: Evolutionary Learning of Globally Optimal Classification and Regression Trees in R." *Working Paper 2011-20*, Working Papers in Economics and Statistics, Research Platform Empirical and Experimental Economics, Universität Innsbruck. URL http://EconPapers.RePEc.org/RePEc:inn:wpaper:2011-20.

Hornik K, Buchta C, Zeileis A (2009). "Open-Source Machine Learning: R Meets **Weka**." *Computational Statistics*, **24**(2), 225–232.

Hothorn T (2014). "CRAN Task View: Machine Learning & Statistical Learning." Version 2014-03-07, URL http://CRAN.R-project.org/view=MachineLearning.

Hothorn T, Hornik K, Strobl C, Zeileis A (2013). ***party**: A Laboratory for Recursive Partytioning*. R package version 1.0-10, URL http://CRAN.R-project.org/package=party.

Hothorn T, Hornik K, Zeileis A (2006). "Unbiased Recursive Partitioning: A Conditional Inference Framework." *Journal of Computational and Graphical Statistics*, **15**(3), 651–674.

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

Loh WY (2014). "50 Years of Classification and Regression Trees." *International Statistical Review*. Forthcoming.

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

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

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

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

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.

Theußl S, Zeileis A (2009). "Collaborative Software Development Using R-Forge." *The R Journal*, **1**(1), 9–14. URL http://journal.R-project.org/.

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

Zeileis A, Hothorn T, Hornik K (2008). "Model-Based Recursive Partitioning." *Journal of Computational and Graphical Statistics*, **17**(2), 492–514.

Affiliation:

Torsten Hothorn
Institut für Sozial- und Präventivmedizin, Abteilung Biostatistik
Universität Zürich
Hirschengraben 84
CH-8001 Zürich, Switzerland
E-mail: Torsten.Hothorn@R-project.org
URL: http://user.math.uzh.ch/hothorn/

Achim Zeileis
Department of Statistics
Faculty of Economics and Statistics
Universität Innsbruck
Universitätsstr. 15
6020 Innsbruck, Austria
E-mail: Achim.Zeileis@R-project.org
URL: http://eeecon.uibk.ac.at/~zeileis/

University of Innsbruck - Working Papers in Economics and Statistics
Recent Papers can be accessed on the following webpage:

http://eeecon.uibk.ac.at/wopec/

2014-10 **Torsten Hothorn, Achim Zeileis:** partykit: A modular toolkit for recursive partytioning in R

2014-09 **Rudi Stracke, Wolfgang Höchtl, Rudolf Kerschbamer, Uwe Sunde:** Incentives and selection in promotion contests: Is it possible to kill two birds with one stone?

2014-08 **Rudi Stracke, Wolfgang Höchtl, Rudolf Kerschbamer, Uwe Sunde:** Optimal prizes in dynamic elimination contests: Theory and experimental evidence

2014-07 **Nikolaos Antonakakis, Max Breitenlechner, Johann Scharler:** How strongly are business cycles and financial cycles linked in the G7 countries?

2014-06 **Burkhard Raunig, Johann Scharler, Friedrich Sindermann:** Do banks lend less in uncertain times?

2014-05 **Julia Auckenthaler, Alexander Kupfer, Rupert Sendlhofer:** The impact of liquidity on inflation-linked bonds: A hypothetical indexed bonds approach

2014-04 **Alice Sanwald, Engelbert Theurl:** What drives out-of pocket health expenditures of private households? - Empirical evidence from the Austrian household budget survey

2014-03 **Tanja Hörtnagl, Rudolf Kerschbamer:** How the value of information shapes the value of commitment or: Why the value of commitment does not vanish

2014-02 **Adrian Beck, Rudolf Kerschbamer, Jianying Qiu, Matthias Sutter:** Car mechanics in the lab - Investigating the behavior of real experts on experimental markets for credence goods

2014-01 **Loukas Balafoutas, Adrian Beck, Rudolf Kerschbamer, Matthias Sutter:** The hidden costs of tax evasion - Collaborative tax evasion in markets for expert services

2013-37 **Reto Stauffer, Georg J. Mayr, Markus Dabernig, Achim Zeileis:** Somewhere over the rainbow: How to make effective use of colors in meteorological visualizations

2013-36 **Hannah Frick, Carolin Strobl, Achim Zeileis:** Rasch mixture models for DIF detection: A comparison of old and new score specifications

2013-35 **Nadja Klein, Thomas Kneib, Stephan Klasen, Stefan Lang:** Bayesian structured additive distributional regression for multivariate responses

2013-34 **Sylvia Kaufmann, Johann Scharler:** Bank-lending standards, loan growth and the business cycle in the Euro area

2013-33 **Ting Wang, Edgar C. Merkle, Achim Zeileis:** Score-based tests of measurement invariance: Use in practice

2013-32 **Jakob W. Messner, Georg J. Mayr, Daniel S. Wilks, Achim Zeileis:** Extending extended logistic regression for ensemble post-processing: Extended vs. separate vs. ordered vs. censored

2013-31 **Anita Gantner, Kristian Horn, Rudolf Kerschbamer:** Fair division in unanimity bargaining with subjective claims

2013-30 **Anita Gantner, Rudolf Kerschbamer:** Fairness and efficiency in a subjective claims problem

2013-29 **Tanja Hörtnagl, Rudolf Kerschbamer, Rudi Stracke, Uwe Sunde:** Heterogeneity in rent-seeking contests with multiple stages: Theory and experimental evidence

2013-28 **Dominik Erharter:** Promoting coordination in summary-statistic games

2013-27 **Dominik Erharter:** Screening experts' distributional preferences

2013-26 **Loukas Balafoutas, Rudolf Kerschbamer, Matthias Sutter:** Second-degree moral hazard in a real-world credence goods market

2013-25 **Rudolf Kerschbamer:** The geometry of distributional preferences and a non-parametric identification approach

2013-24 **Nadja Klein, Michel Denuit, Stefan Lang, Thomas Kneib:** Nonlife ratemaking and risk management with bayesian additive models for location, scale and shape

2013-23 **Nadja Klein, Thomas Kneib, Stefan Lang:** Bayesian structured additive distributional regression

2013-22 **David Plavcan, Georg J. Mayr, Achim Zeileis:** Automatic and probabilistic foehn diagnosis with a statistical mixture model

2013-21 **Jakob W. Messner, Georg J. Mayr, Achim Zeileis, Daniel S. Wilks:** Extending extended logistic regression to effectively utilize the ensemble spread

2013-20 **Michael Greinecker, Konrad Podczeck:** Liapounoff's vector measure theorem in Banach spaces *forthcoming in Economic Theory Bulletin*

2013-19 **Florian Lindner:** Decision time and steps of reasoning in a competitive market entry game *forthcoming in Economics Letters*

2013-18 **Michael Greinecker, Konrad Podczeck:** Purification and independence

2013-17 **Loukas Balafoutas, Rudolf Kerschbamer, Martin Kocher, Matthias Sutter:** Revealed distributional preferences: Individuals vs. teams *forthcoming in Journal of Economic Behavior and Organization*

2013-16 **Simone Gobien, Björn Vollan:** Playing with the social network: Social cohesion in resettled and non-resettled communities in Cambodia

2013-15 **Björn Vollan, Sebastian Prediger, Markus Frölich:** Co-managing common pool resources: Do formal rules have to be adapted to traditional ecological norms? *published in Ecological Economics*

2013-14 **Björn Vollan, Yexin Zhou, Andreas Landmann, Biliang Hu, Carsten Herrmann-Pillath:** Cooperation under democracy and authoritarian norms

2013-13 **Florian Lindner, Matthias Sutter:** Level-k reasoning and time pressure in the 11-20 money request game *published in Economics Letters*

2013-12 **Nadja Klein, Thomas Kneib, Stefan Lang:** Bayesian generalized additive models for location, scale and shape for zero-inflated and overdispersed count data

2013-11 **Thomas Stöckl:** Price efficiency and trading behavior in limit order markets with competing insiders *forthcoming in Experimental Economics*

2013-10 **Sebastian Prediger, Björn Vollan, Benedikt Herrmann:** Resource scarcity, spite and cooperation

2013-09 **Andreas Exenberger, Simon Hartmann:** How does institutional change coincide with changes in the quality of life? An exemplary case study

2013-08 **E. Glenn Dutcher, Loukas Balafoutas, Florian Lindner, Dmitry Ryvkin, Matthias Sutter:** Strive to be first or avoid being last: An experiment on relative performance incentives.

2013-07 **Daniela Glätzle-Rützler, Matthias Sutter, Achim Zeileis:** No myopic loss aversion in adolescents? An experimental note

2013-06 **Conrad Kobel, Engelbert Theurl:** Hospital specialisation within a DRG-Framework: The Austrian case

2013-05 **Martin Halla, Mario Lackner, Johann Scharler:** Does the welfare state destroy the family? Evidence from OECD member countries

2013-04 **Thomas Stöckl, Jürgen Huber, Michael Kirchler, Florian Lindner:** Hot hand belief and gambler's fallacy in teams: Evidence from investment experiments

2013-03 **Wolfgang Luhan, Johann Scharler:** Monetary policy, inflation illusion and the Taylor principle: An experimental study

2013-02 **Esther Blanco, Maria Claudia Lopez, James M. Walker:** Tensions between the resource damage and the private benefits of appropriation in the commons

2013-01 **Jakob W. Messner, Achim Zeileis, Jochen Broecker, Georg J. Mayr:** Improved probabilistic wind power forecasts with an inverse power curve transformation and censored regression

University of Innsbruck

Working Papers in Economics and Statistics

2014-10

Torsten Hothorn, Achim Zeileis

partykit: A modular toolkit for recursive partytioning in R

Abstract
The R package partykit provides a flexible toolkit for learning, representing, summarizing, and visualizing a wide range of tree-structured regression and classification models. The functionality encompasses: (a) basic infrastructure for representing trees (inferred by any algorithm) so that unified print/plot/predict methods are available; (b) dedicated methods for trees with constant fits in the leaves (or terminal nodes) along with suitable coercion functions to create such trees (e.g., by rpart, RWeka, PMML); (c) a reimplementation of conditional inference trees (ctree, originally provided in the party package); (d) an extended reimplementation of model-based recursive partitioning (mob, also originally in party) along with dedicated methods for trees with parametric models in the leaves. Here, a brief overview of the package and its design is given while more detailed discussions of items (a)–(d) are available in vignettes accompanying the package.

ISSN 1993-4378 (Print)
ISSN 1993-6885 (Online)