PALM tree Vignette

Heidi Seibold
University of Zurich

Torsten Hothorn
University of Zurich

Achim Zeileis
University of Innsbruck

Abstract

Both generalised linear models (GLMs) and GLM trees are common and useful methods to analyse a wide variety of data. In GLMs effects are linear whereas in GLM trees they are subgroup-wise linear. PALM trees provides provides a compromise between the two by allowing for subgroup-wise linear effects next to globally linear effects. We show how PALM trees can be applied using the function `palmtree` in the R **partykit** package.

Keywords: model-based recursive partitioning, subgroup analyses, GLM.

1. Overview

The **partykit** package (Hothorn and Zeileis 2015) provides an interface to work with recursive partitioning methods. The two major model classes are conditional inference trees (function `ctree()`, Hothorn, Hornik, and Zeileis 2006) and model-based trees (so far functions `mob()`, `lmtree()` and `glmtree()`, Zeileis, Hothorn, and Hornik 2008). The model-based tree family has now a new member, `palmtree()`.

This vignette introduces when and how PALM trees can be used and how they can be computed in R. In terms of the methodology we focus on the essential and refer to Seibold, Hothorn, and Zeileis (2016) for more details. Section 2 gives a short theoretical introduction followed by two illustrative applications with simple PALM trees. Section 3 shows the more detailed settings that can be made (3.1) and goes into detail of model choices and with this also explains when PALM trees are not needed (3.2).

2. Basic PALM trees

PALM trees are (generalised) linear model trees containing subgroup-wise varying linear effects $\boldsymbol{\beta}(\mathbf{z})$ for some covariates $\mathbf{x}_V$ (V for varying) and additionally globally linear effects $\boldsymbol{\gamma}$ for covariates $\mathbf{x}_F$ (F for fixed). Thus they provide a compromise between (G)LMs where all effects are linear and (G)LM trees where all effects are subgroup dependent. The PALM tree model

$$g(\boldsymbol{\mu}) = \mathbf{x}_V^\top \boldsymbol{\beta}(\mathbf{z}) + \mathbf{x}_F^\top \boldsymbol{\gamma} \tag{1}$$

with expected response $\boldsymbol{\mu} = \mathbb{E}(\mathbf{y})$ and link function g is estimated via an EM-type algorithm that iterates between estimating the model and estimating the tree structure. The tree structure defines the subgroups and is estimated based on split variables $\mathbf{z}$. Hence the varying

parameter vector for each observation i is defined as

$$\boldsymbol{\beta}(\mathbf{z}_i) = \begin{cases} \boldsymbol{\beta}_1 & \text{if } i \text{ in subgroup 1} \\ \boldsymbol{\beta}_2 & \text{if } i \text{ in subgroup 2} \\ \vdots & \end{cases} \tag{2}$$

The algorithm goes as follows:

- Initialize (G)LM with main effects of $\mathbf{x}_V$ and $\mathbf{x}_F$, i.e. $g(\boldsymbol{\mu}) = \mathbf{x}_V^\top \boldsymbol{\beta} + \mathbf{x}_F^\top \boldsymbol{\gamma}$.
- Iterate until convergence of the log-likelihood.
  1. Estimate (G)LM tree while keeping the global effects $\boldsymbol{\gamma}$ fixed by including them as an offset.
  2. Estimate (G)LM by including the tree structure via interaction terms, i.e. $g(\boldsymbol{\mu}) = (\mathbf{x}_V \circ \text{subgroup})^\top \boldsymbol{\beta} + \mathbf{x}_F^\top \boldsymbol{\gamma}$ (with $\mathbf{x}_V \circ$ subgroup interaction term between $\mathbf{x}_V$ and the subgroups).

The estimation of the tree follows the standard (G)LM tree algorithm, i.e. split variables are found via parameter instability tests (possibly with Bonferroni correction) and split points are found via an exhaustive search maximising the sum of likelihoods in the emerging subgroups. We encourage the reader to look at the algorithm in detail, e.g. via

```
> page(palmtree)
```

To estimate a PALM tree in R with the `palmtree()` function one needs to at least specify the `formula` and the `data`:

```
> palmtree(formula, data)
```

The `formula` consists of four parts:

- outcome y,
- covariates with possibly variable effects $\mathbf{x}_V$,
- covariates with globally fixed effects $\mathbf{x}_F$, and
- covariates used for subgroup definition $\mathbf{z}$, i.e.

$$\texttt{formula = y} \sim \overbrace{\texttt{x\_V1 + ...+ x\_Vk}}^{\mathbf{x}_V} \mid \overbrace{\texttt{x\_F1 + ...+ x\_Fq}}^{\mathbf{x}_F} \mid \overbrace{\texttt{z\_1 + ...+ z\_p}}^{\mathbf{z}}$$

In the following we illustrate the usage of PALM trees in two different settings. Section 2.1 shows simulated data according to a clinical trial. Knowing the data generating process helps in understanding when PALM trees are useful. Section **??** shows an application of a PALM tree on data from a mathematics exam where students self selected in two exam groups.

2.1. Simulated data

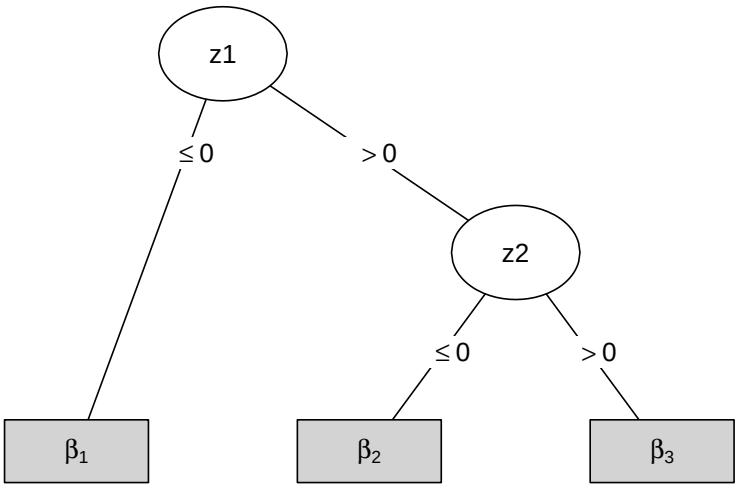


Figure 1: Tree according to data generating process.

We use simulated data that resembles a clinical trial with treatment indicator

$$x_V = \begin{cases} 1 & \text{if treatment is given} \\ 0 & \text{if placebo is given.} \end{cases} \tag{3}$$

30 patient characteristics $\mathbf{z} = (x_1, \ldots, x_{30})$ are simulated from a multivariate normal distribution with correlation 0.2. Patient characteristics x_1 and x_2 are the patient characteristics defining the subgroups given in Figure 1. The treatment effect is defined as

$$\beta(\mathbf{z}) = \begin{cases} -0.375 & = \beta_1 & \text{if } x_1 \leq 0 \\ 0.125 & = \beta_2 = \beta_1 + 0.5 & \text{if } x_1 > 0 \ \wedge \ x_2 \leq 0 \\ 0.625 & = \beta_3 = \beta_2 + 0.5 & \text{if } x_1 > 0 \ \wedge \ x_2 > 0. \end{cases} \tag{4}$$

Patient characteristics x_3 and x_4 are covariates with a direct effect on the primary outcome, i.e. $\mathbf{x}_F = (x_3, x_4)^\top$. Patient characteristics x_5 to x_{30} are noise variables that have no impact on neither the treatment effect $\beta(\mathbf{z})$ nor the primary outcome. Accordingly we simulate the primary outcome (a health score) $\mathbf{y}$ with

$$\begin{aligned} \mathbf{y} =& x_V \beta(\mathbf{z}) + \mathbf{x}_F \boldsymbol{\gamma} + \boldsymbol{\epsilon} \\ =& I(x_1 \leq 0) x_V \beta_1 + \\ & I(x_1 > 0 \wedge x_2 \leq 0) x_V \beta_2 + \\ & I(x_1 > 0 \wedge x_2 > 0) x_V \beta_3 + \\ & \mathbf{x}_F \boldsymbol{\gamma} + \boldsymbol{\epsilon} \end{aligned} \tag{5}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, 1.5)$ is the error term. The following R function can be used to generate the described data:

```
> dgp <- function() {
+
+   nobs <- 1000
+   npc <- 30
+
+   ## patient characteristics
+   x <- mvtnorm::rmvnorm(nobs, mean = rep(0, npc),
+                         sigma = diag(1 - 0.2, npc) + 0.2)
+   colnames(x) <- paste0("x", 1:npc)
+   d <- as.data.frame(x)
+
+   ## treatment xV
+   d$xV<- rbinom(nobs, size = 1, prob = 0.5)
+
+   ## error term
+   d$err <- rnorm(nobs, mean = 0, sd = 1.5)
+
+   ## predictive and prognostic factors
+   which_pred <- 1:2
+   which_prog <- 3:4
+
+   ## define subgroups
+   rules <- t(t(x[, which_pred]) > c(0, 0))
+   d$group <- 1
+   d$group[rules[, 1] == 1] <- 2
+   d$group[rowSums(rules) == 2] <- 3
+   d$group <- as.factor(d$group)
+
+   ## response function mu
+   eff_trt <- c(-0.375, 0.125, 0.625)
+   modelmat <- model.matrix(~ group - 1, data = d)
+   d$trt_effect <- modelmat %*% eff_trt
+   d$mu0 <- as.vector(x[ , which_prog] %*% c(1, 1))
+   d$mu1 <- as.vector(d$mu0 + d$trt_effect)
+   idmu <- cbind(seq_len(nrow(d)), d$xV + 1)
+   d$mu <- d[ , c("mu0", "mu1")][idmu]
+
+   ## outcome y
+   d$y <- d$mu + d$err
+
+   d$xV <- factor(d$xV)
+   return(d)
+ }
```

The results of a simulation study using this data generating process and variations thereof can be found in Seibold *et al.* (2016).

In order to apply a PALM tree to a data set simulated from this data generating process, we

first need to generate a data set:

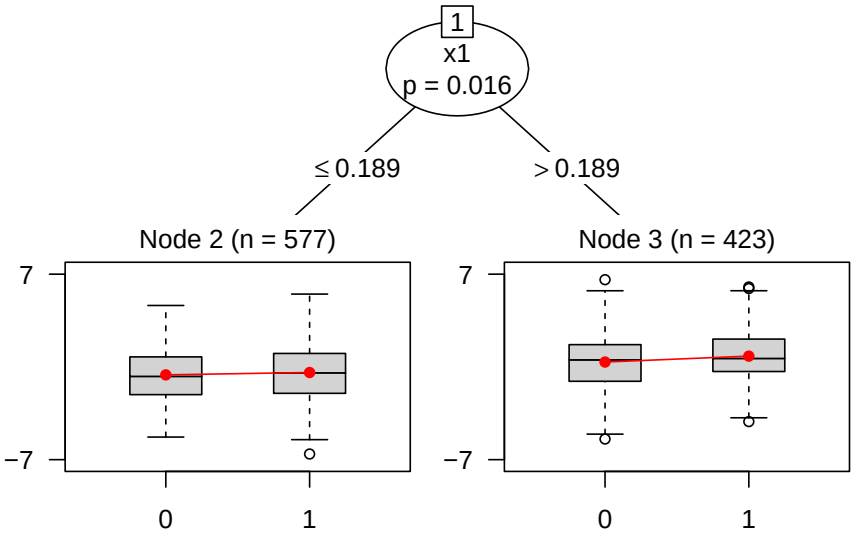
```
> set.seed(123)
> data_sim <- dgp()
```

Next we construct the formula and estimate and plot the PALM tree

```
> x_sim <- paste0("x", 1:30)
> fmla_sim <- as.formula(
+   paste("y ~ xV | x3 + x4 |",
+         paste(x_sim, collapse = " + "))
+ )
> library("palmtree")
> (palmtree_sim <- palmtree(fmla_sim, data = data_sim))
```

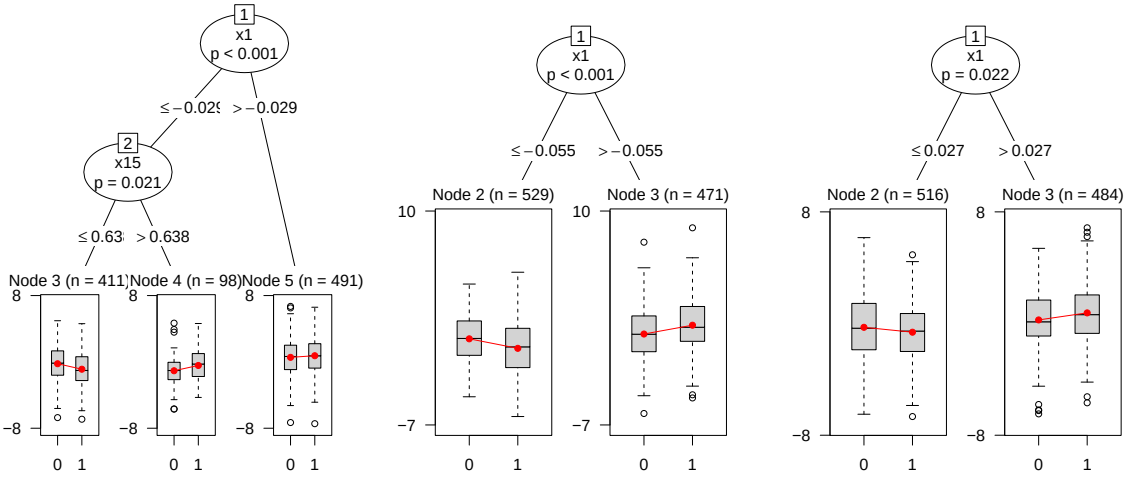
```
Partially additive linear model tree

Model formula:
y ~ xV | x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 +
    x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 + x20 + x21 +
    x22 + x23 + x24 + x25 + x26 + x27 + x28 + x29 + x30

Fitted party:
[1] root
|   [2] x1 <= 0.18904: n = 577
|       (Intercept)         xV1
|        -0.1396067  -0.1021330
|   [3] x1 > 0.18904: n = 423
|       (Intercept)         xV1
|          0.100167    0.245325

Number of inner nodes:    1
Number of terminal nodes: 2
Number of parameters per node: 2
Objective function (residual sum of squares): 2053.511

Linear fixed effects (from palm model):
      x3       x4
1.024749 1.027163
```

```
> plot(palmtree_sim)
```



Note that if we redraw a new data set from the same data generating process, we get different results.

```
> set.seed(222)
> for(i in 1:3) {
+   data_sim_i <- dgp()
+   palmtree_sim_i <- palmtree(fmla_sim, data = data_sim_i)
+   plot(palmtree_sim_i)
+ }
```



2.2. Mathematics exam

We analyse the data of the first-year mathematics exam of business and economics students at the University of Innsbruck in the fall semester 2014/15. The data can be accessed via

```
> data("MathExam14W", package = "psychotools")
> ## scale points achieved to [0, 100] percent
> MathExam14W$tests <- 100 * MathExam14W$tests/26
> MathExam14W$pcorrect <- 100 * MathExam14W$nsolved/13
> ## select variables to be used
> MathExam <- MathExam14W[ , c("pcorrect", "group", "tests", "study",
+                              "attempt", "semester", "gender")]
```

Due to the large number of students (729) the students were asked to select a group, where the first group worte the exam in the morning and the second group right after the first group finished. The students in the two groups received slightly different tasks. The variable `group` contains the information on the selected group of each students. We are interested in whether the exam was fair in the sense that both groups performed similarly in the exam. The performance is measure in percentage of correctly answered questions (`pcorrect`). To account for possibly varying skills in the two groups the performance during the semester which was measured by biweekly online tests (`tests`) can be used. Further student characteristics were obtained, which are the type of `study` (three year bachelor program vs. four year diploma program), the number of times the student has already attempted the exam (`attempt`), the `semester` the student is in and the `gender`. A summary of the data is given below:

```
> summary(MathExam)
```

```
    pcorrect      group       tests         study     attempt    semester
 Min.   :  0.00   1:334   Min.   : 34.62   155:146   1:431   Min.   : 1.00
 1st Qu.: 38.46   2:395   1st Qu.: 65.38   571:583   2: 52   1st Qu.: 1.00
 Median : 53.85           Median : 76.92             3:121   Median : 1.00
 Mean   : 56.34           Mean   : 75.48             4:113   Mean   : 3.11
 3rd Qu.: 69.23           3rd Qu.: 88.46             5: 12   3rd Qu.: 5.00
 Max.   :100.00           Max.   :100.00                     Max.   :21.00
    gender
 female:326
 male  :403
```

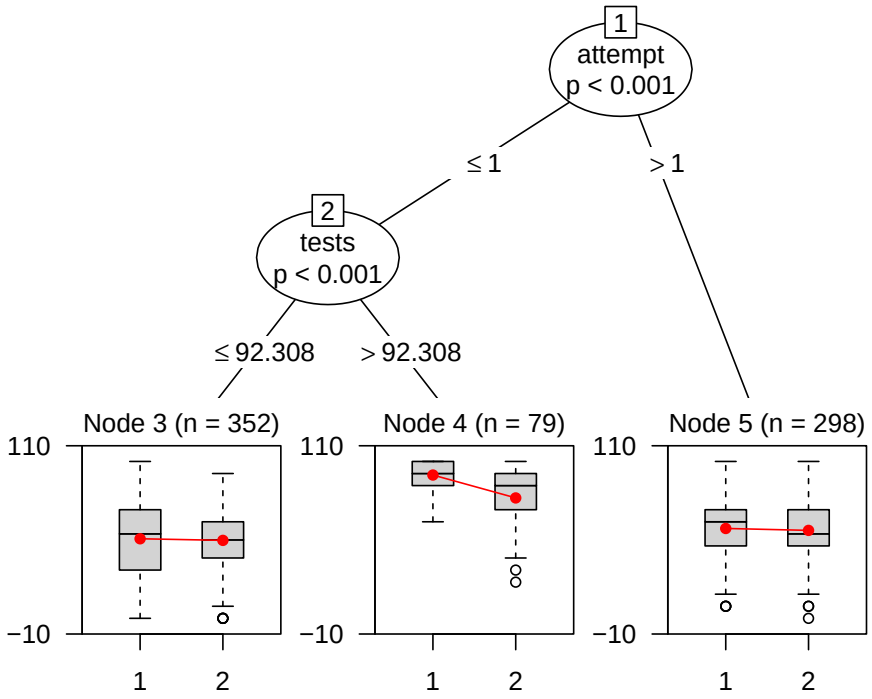
A PALM tree for the mathematics exam data can be estimated via

```
> (palmtree_math <- palmtree(pcorrect ~ group | tests | tests + study +
+                              attempt + semester + gender, data = MathExam))

Partially additive linear model tree

Model formula:
pcorrect ~ group | tests + study + attempt + semester + gender

Fitted party:
[1] root
|   [2] attempt <= 1
|   |   [3] tests <= 92.30769: n = 352
```

```
|   |       (Intercept)      group2
|   |         -7.088499   -2.997321
|   |   [4] tests > 92.30769: n = 79
|   |       (Intercept)      group2
|   |          13.98085   -14.49418
|   [5] attempt > 1: n = 298
|       (Intercept)      group2
|          2.332298   -1.704129

Number of inner nodes:    2
Number of terminal nodes: 3
Number of parameters per node: 2
Objective function (residual sum of squares): 253218

Linear fixed effects (from palm model):
    tests
0.7868941
```

The plot

```
> plot(palmtree_math)
```



reaveals that we need to differentate between students who attempt the exam for the first time and students who have attempted the exam before. For the studentes who attempt the

exam for the first time, we need to differentiate between student who scored very high (more than 92.308 %) in the online tests that were written during the semester and the students who did not score as high.

To obtain coefficients from the PALM tree there are three different options

```
> coef(palmtree_math)

  (Intercept)        .tree4        .tree5         tests .tree3:group2
   -7.0884991    21.0693481     9.4207969     0.7868941    -2.9973211
.tree4:group2 .tree5:group2
  -14.4941767    -1.7041287

> coef(palmtree_math$palm)

  (Intercept)        .tree4        .tree5         tests .tree3:group2
   -7.0884991    21.0693481     9.4207969     0.7868941    -2.9973211
.tree4:group2 .tree5:group2
  -14.4941767    -1.7041287

> coef(palmtree_math$tree)

  (Intercept)     group2
3   -7.088499  -2.997321
4   13.980849 -14.494177
5    2.332298  -1.704129
```

where the first two are equivalent. They return coefficients of the model

```
> (palmmod1 <- lm(pcorrect ~ .tree + group:.tree + tests,
+                 data = palmtree_math$data))

Call:
lm(formula = pcorrect ~ .tree + group:.tree + tests, data = palmtree_math$data)

Coefficients:
  (Intercept)         .tree4         .tree5          tests  .tree3:group2
      -7.0885        21.0693         9.4208         0.7869        -2.9973
.tree4:group2  .tree5:group2
     -14.4942        -1.7041
```

whereas the third option returns coefficients of the model

```
> (palmmod2 <- lm(pcorrect ~ 0 + .tree + group:.tree + tests,
+                 data = palmtree_math$data))
```

```
Call:
lm(formula = pcorrect ~ 0 + .tree + group:.tree + tests, data = palmtree_math$data)

Coefficients:
       .tree3         .tree4         .tree5          tests  .tree3:group2
      -7.0885        13.9808         2.3323         0.7869        -2.9973
.tree4:group2  .tree5:group2
     -14.4942        -1.7041
```

Hence the difference is in the sense that `palmmod1` estimates a model with intercept, which in this case can be interpreted as the expected percentage points in the exam for a student in node 3 (first subgroup), who has no correct answers in the online tests and self selected into exam group 1. The effects denoted by `.tree4` and `.tree5` give how many percentage points more are expected for a student who has no correct answers in the online tests and self selected into exam group 1 if she is in node 4 (second subgroup) or 5 (third subgroup) respectively. In contrast `palmmod2` estimates a model without intercept and hence the effects denoted by `.tree3`, `.tree4` and `.tree5` give the expected percentage points for a student who has no correct answers in the online tests and self selected into exam group 1 within the three subgroups.

3. Advanced PALM trees

3.1. Settings for PALM trees

Additional to the arguments `formula` and `data` there are several other arguments that can be set in the `palmtree()` function.

```
> palmtree(formula, data, weights = NULL, family = NULL,
+          lmstart = NULL, abstol = 0.001, maxit = 100,
+          dfsplit = TRUE, verbose = FALSE, plot = FALSE, ...)
```

The `family` argument can be used to compute PALM trees for models of the generalised linear model family. If it is NULL a linear model will be computed. The argument `lmstart` allows for different initialisation of the algorithm than with the global model, which is computed using the first three parts of the PALM tree formula, i.e. `y ~ x_v1 + ...+ x_vk + x_f1 + ...+ x_fq`. For the math exam PALM tree computed above, this is global model is

```
> lm(pcorrect ~ group + tests, data = MathExam)
```

Convergence conditions and maximum number of iterations are given by arguments `abstol` and `maxit`. The ellipsis (`...`) are further arguments that are passed on to `mob_control()`, which control the tree growing algorithm such as turning off Bonferroni correction or regulating tree and node size.

3.2. Model design

The decision on whether to use a (G)LM, a (G)LM tree or a PALM tree strongly impacts the result of the analysis. The decision between a (G)LM and a model-based tree should be based on the believe in subgroups where the effect between subgroups differ. If one decides on a model-based tree for modelling data, there are two further model design decisions that have to be made. The first being whether there is the need to differentiate between a varying and a fixed model part and, if one decides that the differentiation is needed, which covariates of the model shall be part of the fixed and which shall be part of the varying model part. The first is essentially the decision between `glmtree` or `lmtree` and `palmtree`. Do we belive that there are covariates that merely have a direct linear effect on the outcome? If so, we can limit the number of paramters and estimate globally fixed effects for these covariates. The second decision should be already contained in the first decision. If we believe that certain covariates have a direct linear effect on the outcome, then it should be already clear which those are.

In the case of a clinical trial subgroup analysis we are primarily interested in whether different patients react differently to the same treatment. There are several considerations to be made:

- Are predictive factors known?
- Is the tree structure known?
- Are prognostic factors known?

Predictive factors are patient characteristics that have an impact on the relation between treatment and primary endpoint, i.e. on the treatment effect. Prognostic factors are patient characteristics that have an effect on the primary enpoint. Below we list some of the computations that can be performed for possible answers to these questions in the case of the simulated data present.

- Everything known

```
> lm(y ~ x3 + x4 +
+      xV * I(x1 > 0 & x2 <= 0) +
+      xV * I(x1 > 0 & x2 > 0), data = data_sim)
```

- Tree structure unknown, relevant factors known, no relevant unknown factors

```
> palmtree(y ~ xV | x3 + x4 | x1 + x2, data = data_sim)
```

- Tree structure unknown, relevant factors known, known prognostic factors known to be linear and not additionally predictive, possibly relevant unknown factors

```
> palmtree(y ~ xV | x3 + x4 | x1 + x2 + x5 + x6 + x7 + x8 + x9 +
+            x10 + x11 + x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 +
+            x20 + x21 + x22 + x23 + x24 + x25 + x26 + x27 + x28 + x29 +
+            x30, data = data_sim)
```

- Tree structure unknown, relevant factors known, known prognostic factors may not be linear or additionally predictive, possibly relevant unknown factors

```
> lmtree(y ~ xV + x3 + x4 | x1 + x2 + x5 + x6 + x7 + x8 + x9 +
+            x10 + x11 + x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 +
+            x20 + x21 + x22 + x23 + x24 + x25 + x26 + x27 + x28 + x29 +
+            x30, data = data_sim)
```

- Tree structure unknown, relevant factors known, known prognostic factors could also be predictive, possibly relevant unknown factors

```
> palmtree(y ~ xV | x3 + x4 | x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 +
+            x10 + x11 + x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 +
+            x20 + x21 + x22 + x23 + x24 + x25 + x26 + x27 + x28 + x29 +
+            x30, data = data_sim)
```

- Relevant prognostic factors unknown

```
> lmtree(y ~ xV | x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 +
+          x10 + x11 + x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 +
+          x20 + x21 + x22 + x23 + x24 + x25 + x26 + x27 + x28 + x29 +
+          x30, data = data_sim)
```

4. Conclusion

References

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

Hothorn T, Zeileis A (2015). "partykit: A Modular Toolkit for Recursive Partytioning in R." *Journal of Machine Learning Research*, **16**, 3905–3909. URL http://jmlr.org/papers/v16/hothorn15a.html.

Seibold H, Hothorn T, Zeileis A (2016). "Generalised Linear Model Trees with Global Additive Effects." *ArXiv e-prints*. URL https://arxiv.org/abs/1612.07498.

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

Affiliation:

Heidi Seibold
Department of Biostatistics
Epidemiology, Biostatistics and Prevention Institute
University of Zurich
Hirschengraben 84
CH-8001 Zurich, Switzerland
E-mail: Heidi.Seibold@uzh.ch
URL: http://www.ebpi.uzh.ch/en/aboutus/departments/biostatistics/teambiostats/seibold.html

Torsten Hothorn
Department of Biostatistics
Epidemiology, Biostatistics and Prevention Institute
University of Zurich
Hirschengraben 84
CH-8001 Zurich, 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
University of Innsbruck
Universitätsstr. 15
AT-6020 Innsbruck, Austria
E-mail: Achim.Zeileis@R-project.org
URL: http://eeecon.uibk.ac.at/~zeileis/