Package ‘MplusTrees’

March 31, 2025

Type Package

Title Decision Trees with Structural Equation Models Fit in 'Mplus'

Version 0.2.3

Maintainer Sarfaraz Serang <sserang@mailbox.sc.edu>

Depends R (>= 2.10), rpart, MplusAutomation

Imports nlme, rpart.plot

Suggests lavaan

Description
Uses recursive partitioning to create homogeneous subgroups based on structural equation models fit in 'Mplus', a stand-alone program developed by Muthen and Muthen.

SystemRequirements 'Mplus' (<http://www.statmodel.com>)

License GPL

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Sarfaraz Serang [aut, cre],
Ross Jacobucci [aut],
Kevin J. Grimm [ctb],
Gabriela Stegmann [ctb],
Andreas M. Brandmaier [ctb]

Repository CRAN

Date/Publication 2025-03-31 18:10:02 UTC

Contents

causalmpt *Causal Mplus Trees*

Description

Uses M*plus* Trees to match on structural equation model parameters in matching subsample. Then estimates Conditional Average Treatment Effects (CATEs) in holdout estimation subsample.

Usage

```
causalmpt(
  script,
  data,
  rPartFormula,
  group = ~id,
  treat,
  outcome,
  est.samp = 0.2,
  ...
)
```

Arguments

`script`	An `MplusAutomation` script file
`data`	Dataset that is specified in the script
`rPartFormula`	Formula of the form ~ variable names
`group`	id variable. If not specified an id variable is created for each row
`treat`	Treatment variable
`outcome`	Univariate outcome of interest (dependent variable in mean comparison tests)
`est.samp`	Proportion of sample to be used as holdout sample (estimation subsample)
`...`	Other arguments to `MplusTrees` for building Mplus Tree

Details

See documentation for `MplusTrees()` for further information on tree building process. Takes terminal nodes from Mplus Tree and considers them "matched". Splits estimation subsample into groups defined by covariate pattern in terminal nodes from Mplus Tree. Performs t tests in each group with `treat` as independent variable and `outcome` as dependent variable to estimate CATEs. Also performs ANOVA to determine if treatment effect differs by group (interaction).

Value

An object of class '`causalmpt`'. Tree structure drawn from `MplusTrees()`. CATEs estimated in estimation (holdout) subsample. Provides results of t tests to estimate CATEs in each group and ANOVA to examine group differences in treatment effect.

Author(s)

Sarfaraz Serang

References

Serang, S., & Sears, J. (2021). Tree-based matching on structural equation model parameters. Behavioral Data Science, 1, 31-53.

Examples

```
## Not run:
library(lavaan)

script = mplusObject(
   TITLE = "Causal Mplus Trees Example",
   MODEL = "f1 BY x1-x3;",
   usevariables = c('x1','x2','x3'),
   rdata = HolzingerSwineford1939)

fit.cmpt = causalmpt(script, HolzingerSwineford1939, group=~id,
   rPartFormula=~school+grade,
   control=rpart.control(minsplit=100, minbucket=100, cp=.01),
   treat="sex", outcome="x4")
fit.cmpt

## End(Not run)
```

MplusTrees *Recursive partitioning trees with Mplus models*

Description

Generates recursive partitioning trees using M*plus* models. `MplusTrees()` takes an M*plus* model written in the form of an `MplusAutomation` script, uses `MplusAutomation` to fit the model in M*plus*, and performs recursive partitioning using `rpart`.

Usage

```
MplusTrees(
  script,
  data,
  rPartFormula,
  catvars = NULL,
  group = ~id,
  control = rpart.control(),
  se = F,
  psplit = F,
  palpha = 0.05,
  cv = F,
  k = 5
)
```

Arguments

`script`	An `MplusAutomation` script file
`data`	Dataset that is specified in the script
`rPartFormula`	Formula of the form ~ variable names
`catvars`	Vector of names of categorical covariates
`group`	id variable. If not specified an id variable is created for each row
`control`	Control object for `rpart`
`se`	Whether to print standard errors and *p* values. In general should be set to FALSE
`psplit`	Whether to use likelihood ratio *p* values as a splitting criterion
`palpha`	Type I error rate (alpha level) for rejecting with likelihood ratio test when `psplit` set to `TRUE`
`cv`	Performs k-fold cross-validation to select value of `cp`
`k`	number of folds for cross-validation

Details

The function temporarily changes the working directory to the temporary directory. Files used and generated by M*plus* are stored here and can be accessed using `tempdir()`.

By default `MplusTrees()` only splits on the criteria specified in the `control` argument, the most important of which is the `cp` parameter. The user can also split on the *p* value generated from the likelihood ratio test comparing the parent node to a multiple group model consisting of 2 groups (the daughter nodes). This *p* value criterion is used in addition to the `cp` criterion in that both must be met for a split to be made. The `psplit` argument turns this option on, and `palpha` sets the alpha level criterion for rejection.

Cross-validation (CV) can also be used to choose the `cp` parameter. If this option is used, any user-specified `cp` value will be overridden by the optimal `cp` value chosen by CV. CV fits the model to the training set and calculates an expected minus 2 log-likelihood (-2LL) for each terminal node. In the test set, individuals are assigned to terminal nodes based on the tree structure found in the training set. Their "expected" values are the -2LL values from the respective training set terminal nodes. The "observed" values are the -2LL values from fitting a multiple group model, with each terminal node as a group. The `cp` value chosen is the one that produces the smallest MSE.

CV should only be used when (1) the M*plus* model can be fit relatively quickly, (2) there are only a few covariates with a few response options, and (3) the sample size is large enough that the user is confident the model can be fit without issue in a sample of size *N/k* and a tree that partitions this sample further. If these conditions are not met, the process could take prohibitively long to arrive at a solution. Note that if even a single model fails to produce a valid log-likelihood value, the function will terminate with an error.

Value

An object of class 'mplustree'. rpart_out provides the tree structure, terminal gives a vector of terminal nodes, where shows the terminal node of each id, and estimates gives the parameter estimates for each terminal node.

Author(s)

Ross Jacobucci and Sarfaraz Serang

References

Serang, S., Jacobucci, R., Stegmann, G., Brandmaier, A. M., Culianos, D., & Grimm, K. J. (2021). Mplus Trees: Structural equation model trees using Mplus. Structural Equation Modeling, 28, 127-137.

Examples

```
## Not run:
library(lavaan)

script = mplusObject(
   TITLE = "Example #1 - Factor Model;",
   MODEL = "f1 BY x1-x3; f2 BY x4-x6; f3 BY x7-x9;",
   usevariables = c('x1','x2','x3','x4','x5','x6','x7','x8','x9'),
   rdata = HolzingerSwineford1939)

fit = MplusTrees(script, HolzingerSwineford1939, group=~id,
   rPartFormula=~sex+school+grade,
   control=rpart.control(minsplit=100, minbucket=100, cp=.01))

fit

## End(Not run)
```

plot.mplustree *Plots tree structure of an Mplus Tree*

Description

Wrapper using rpart.plot package to plot the tree structure of a fitted Mplus Tree.

Usage

```
## S3 method for class 'mplustree'
plot(x, ...)
```

Arguments

x	An object of class "mplustree" (a fitted Mplus Tree)
...	Other arguments passed to rpart.plot

Details

Each node of the plot by default contain the -2 log-likelihood (deviance), the number of individuals in the node, and the percentage of the total sample in the node.

Author(s)

Sarfaraz Serang, relying heavily on the rpart.plot package by Stephen Milborrow.

Examples

```
## Not run:
library(lavaan)

script = mplusObject(
   TITLE = "Example #1 - Factor Model;",
   MODEL = "f1 BY x1-x3; f2 BY x4-x6; f3 BY x7-x9;",
   usevariables = c('x1','x2','x3','x4','x5','x6','x7','x8','x9'),
   rdata = HolzingerSwineford1939)

fit = MplusTrees(script, HolzingerSwineford1939, group=~id,
   rPartFormula=~sex+school+grade, control=rpart.control(cp=.01))

fit

plot(fit)

## End(Not run)
```

---

summary.mplustree *Summarizing MplusTrees model Fits*

Description

summary method for class "mplustree".

Usage

```
## S3 method for class 'mplustree'
summary(object, ...)
```

Arguments

object	An object of class "mplustree" (a fitted Mplus Tree)
...	Other arguments passed to or from other methods

Details

Prints the tree structure given in `object`

Examples

```
## Not run:
library(lavaan)

script = mplusObject(
   TITLE = "Example #1 - Factor Model;",
   MODEL = "f1 BY x1-x3; f2 BY x4-x6; f3 BY x7-x9;",
   usevariables = c('x1','x2','x3','x4','x5','x6','x7','x8','x9'),
   rdata = HolzingerSwineford1939)

fit = MplusTrees(script, HolzingerSwineford1939, group=~id,
   rPartFormula=~sex+school+grade, control=rpart.control(cp=.01))

summary(fit)

## End(Not run)
```

Index