Package 'gmediation'

June 27, 2017

Type Package

Title Mediation Analysis for Multiple and Multi-Stage Mediators

Version 0.1.1

Author Jang Ik Cho, Jeffrey Albert

Maintainer Jang Ik Cho <jxc864@case.edu>

Description Current version of this R package conducts mediation path analysis for multiple mediators in two stages.

License GPL-2 | GPL-3

Encoding UTF-8

LazyLoad true

LazyData true

Imports plyr (>= 1.8.4), VGAM (>= 1.0), MASS

Suggests mediation, knitr, rmarkdown

RoxygenNote 6.0.1

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2017-06-27 14:38:01 UTC

R topics documented:

---

dental   *gmediate example : Dental Data*

Description

To assess paths between socioeconomic status and dental caries in children

Format

data frame

---

gmediate   *Generalized Causal Mediation and Path Analysis*

Description

This function estimates causal mediation and path effects for models with two stages of mediation

Usage

```
gmediate(data, model.m1, model.m2, model.y, expos, ref = NULL, refmult = 1,
  seed = 1234, bootsims = 500, cluster = NULL, decomp = 1,
  sig.level = 95, sens.par = c(-0.9, 0, 0.5, 0.9))
```

Arguments

data	a data frame
model.m1	a list of the model fit from each first stage mediators
model.m2	a list of the model fit from each second stage mediators
model.y	an object of the model fit from the response variable
expos	a vector with the name of the exposure variable and exposure value e.g. c("exposure variable name", c(exposure value))
ref	a vector with the name of the reference group variable and reference group value e.g. c("reference group variable name", c(reference group value))
refmult	numeric value used as a multiplier to increase the number of monte carlo draws
seed	seed to generate bootstrap samples
bootsims	number of bootstrap samples to generate
cluster	the cluster variable that will be used for cluster bootstrap resampling
decomp	option with two choices for decomposition of total exposure effect
sig.level	confidence level for bootstrap confidence interval
sens.par	a vector with numeric values for sensitivity parameters to be used when the path is non-identifiable

Details

This function carries out inference for mediation (path) effects based on user-specified models. The function allows for two stages of mediation (up to three links from exposure/treatment, X, to final outcome, Y), and an arbitrary number of mediators at each stage. The user specifies an allowed generalized regression model for each outcome (Y and each mediator). Allowed distributions (link functions) include: normal (identity link), Bernoulli (logit link), Poisson (log link), and negative binomial (log link); note that canonical links are used for each distribution. Unsaturated models (i.e., leaving out earlier stage mediators) are allowed, possibly imposing a prior null paths. Baseline (pre-exposure) covariates may be included in each model. The user-specified models are re-fit in the function using the complete cases (based on all the model variables).

The user may also specify a reference group, effectively defining the empirical multivariate baseline covariate distribution used in the mediation formula; this may be viewed as defining the sub-population to which inference is performed. The default reference group is the whole (complete case) sample. The reference group may also be 'cloned' by a specified integer multiple ("refmult") to reduce Monte Carlo error (at the possible cost of longer computing time).

An extended mediation formula is used to estimate each path effect. The sum of effects (or 'overall effect') for paths through each mediator is also computed. For paths (and overall) effects that are 'non-identifiable', multiple estimates are provided according to specified sensitivity parameters values ("sens.par", a list of values between – 1 and 1), the default values being -0.9, 0, 0.5, and 0.9. Path effects have alternative versions depending on the manner in which the total exposure effect is decomposed. Two choices, using "decomp", are provided; the ordering of mediators in specified models also generally affect the decomposition.

Bootstrap resampling is used to obtain bootstrap percentile method confidence intervals (at the 95 percent, or user-specified, confidence level). The option "cluster" can be used to specify a unit (aside from the individual, that is, each data record, which is the default) for which cluster bootstrap resampling is performed.

See Also

mediation

Examples

```
library(gmediation)

## mediation analysis for two stage mediators with one mediator at each stage
model.y1 <- glm(dmftd ~ ses + grpvlb + grpbpd + race + SEX + brush + seal, family = binomial,
                data = dental)
model.m11 <- glm(brush ~ ses + grpvlb + grpbpd + race + SEX, family = binomial, data = dental)
model.m21 <- glm(seal ~ ses + grpvlb + grpbpd + race + SEX + brush, family = binomial,
                data = dental)
single <- gmediate(data = dental, model.m1 = list(model.m11), model.m2 = list(model.m21),
              model.y = model.y1, expos = c("ses", 1), ref = c("ses", c(0,1)), seed = 1234,
              bootsims = 100, cluster = NULL, decomp = 1, sig.level = 95, sens.par = c(0),
                  refmult = 1)
single
names(single)

## Not run:
```

```
## mediation analysis for two stage mediators with two mediator at each stage
model.y2 <- glm(dmftd ~ ses + grpvlb + grpbpd + race + SEX + brush + visit + seal + ohi,
                family = binomial, data = dental)
model.m11 <- glm(brush ~ ses + grpvlb + grpbpd + race + SEX, family = binomial, data = dental)
model.m12 <- glm(visit ~ ses + grpvlb + grpbpd + race + SEX, family = binomial, data = dental)
model.m21 <- glm(seal ~ ses + grpvlb + grpbpd + race + SEX + brush + visit, family = binomial,
                data = dental)
model.m22 <- glm(ohi ~ ses + grpvlb + grpbpd + race + SEX + brush + visit, family = gaussian,
                data = dental)
gmediate(data = dental, model.m1 = list(model.m11,model.m12), model.m2 = list(model.m21,model.m22),
         model.y = model.y2, expos = c("ses", 1), ref = c("ses", c(0,1)), seed = 352,
         bootsims = 100, cluster = NULL, decomp = 2, sig.level = 95, sens.par = c(0,0.9),
         refmult = 2)

## End(Not run)
```

Index