Package 'mediation'

January 2, 2012

Version 3.1.2

Date 2011-09-19

Title R Package for Causal Mediation Analysis

Author Dustin Tingley <dtingley@gov.harvard.edu>, Teppei Yamamoto <teppei@mit.edu>, Luke Keele <keele.4@polisci.osu.edu>, Kosuke Imai <kimai@princeton.edu>

Maintainer Dustin Tingley <dtingley@gov.harvard.edu>

Depends R (>= 2.7), MASS, Matrix

Suggests mgcv, quantreg, VGAM, sandwich, SuppDists, arm

Description mediation is a publicly available R package that allows both parametric and nonparametric causal mediation analysis. It implements the methods and suggestions in Imai, Keele, and Yamamoto (2010) and Imai, Keele, Tingley (2010). In addition to the estimation of causal mediation effects, the software also allows researchers to conduct sensitivity analysis for certain parametric models.

LazyLoad yes

LazyData yes

License GPL (>= 2)

Repository CRAN

Date/Publication 2011-09-20 08:30:11

R topics documented:

`jobs` *JOB II data*

Description

Job Search Intervention Study (JOBS II). JOBS II is a randomized field experiment that investigates the efficacy of a job training intervention on unemployed workers. The program is designed to not only increase reemployment among the unemployed but also enhance the mental health of the job seekers. In the JOBS II field experiment, 1,801 unemployed workers received a pre-screening questionnaire and were then randomly assigned to treatment and control groups. Those in the treatment group participated in job-skills workshops. In the workshops, respondents learned job-search skills and coping strategies for dealing with setbacks in the job-search process. Those in the control condition received a booklet describing job-search tips. In follow-up interviews, the two key outcome variables were measured; a continuous measure of depressive symptoms based on the Hopkins Symptom Checklist, and a binary variable, representing whether the respondent had become employed.

Usage

```
data
```

Format

A data matrix containing no missing values and hence is provided only for illustrative purposes and not for inference about program efficacy.

Source

The complete JOBS II data is available from the data archives at www.icpsr.umich.edu/

References

Vinokur, A. and Schul, Y. (1997). Mastery and inoculation against setbacks as active ingredients in the jobs intervention for the unemployed. Journal of Consulting and Clinical Psychology 65, 5.

mediate *Causal Mediation Analysis*

Description

'mediate' is used to estimate various quantities for causal mediation analysis, including average causal mediation effects (indirect effect), average direct effects, proportions mediated, and total effect.

Usage

```
mediate(model.m, model.y, sims=1000, boot=FALSE,
        treat="treat.name", mediator="med.name",
        control=NULL, conf.level=.95,
        control.value=0, treat.value=1,
        long=TRUE, dropobs=FALSE, robustSE=FALSE, cluster=NULL, ...)
```

Arguments

model.m	a fitted model object for mediator. Can be of class 'lm', 'polr', 'bayespolr', 'glm', 'gam', or 'rq'.
model.y	a fitted model object for outcome. Can be of class 'lm', 'polr', 'bayespolr', 'glm', 'gam', 'vglm', or 'rq'.
sims	number of Monte Carlo draws for nonparametric bootstrap or quasi-Bayesian approximation.
boot	a logical value. if 'FALSE' a quasi-Bayesian approximation is used for confidence intervals; if 'TRUE' nonparametric bootstrap will be used. Default is 'FALSE'.
conf.level	level of the returned two-sided confidence intervals. Default is to return the 2.5 and 97.5 percentiles of the simulated quantities.
treat	a character string indicating the name of treatment variable used in the models. The treatment can be either binary (integer or a two-valued factor) or continuous (numeric).
mediator	a character string indicating the name of mediator variable used in the models.
control	a character string indicating the name of control group indicator. Only relevant if 'model.y' is of class 'gam'. If provided, 'd0', 'z0' and 'n0' are allowed to differ from 'd1', 'z1' and 'n1', respectively.
control.value	value of the treatment variable used as the control condition. Only relevant when 'treat' is continuous. Default is 0.
treat.value	value of the treatment variable used as the treatment condition. Only relevant when 'treat' is continuous. Default is 1.

long	a logical value. If 'TRUE', the output will contain the entire sets of simulation draws of the the average causal mediation effects, direct effects, proportions mediated, and total effect. Default is 'TRUE'.
dropobs	a logical value indicating the behavior when the model frames of 'model.m' and 'model.y' are composed of different observations. If 'TRUE', models will be re-fitted using the intersection of the two data frames. If 'FALSE', error is returned. Default is 'FALSE'.
robustSE	a logical value. If 'TRUE', heteroskedasticisy-consistent standard errors will be used in quasi-Bayesian simulations. Ignored if 'boot' is 'TRUE' or neither 'model.m' nor 'model.y' has a method for vcovHC in the sandwich package. Default is 'FALSE'.
cluster	a character string indicating the name of a variable to be used for cluster-robust standard errors.
...	other arguments passed to vcovHC in the sandwich package: typically the 'type' argument. Ignored if 'robustSE' is 'FALSE'.

Details

This is the workhorse function for estimating causal mediation effects for a variety of data types. The average causal mediation effect (ACME) represents the expected difference in the potential outcome when the mediator took the value that would realize under the treatment condition as opposed to the control condition, while the treatment status itself is held constant. That is,

$$\delta(t) \;=\; E\{Y(t, M(t_1)) - Y(t, M(t_0))\},$$

where t, t_1, t_0 are particular values of the treatment T such that $t_1 \neq t_0$, $M(t)$ is the potential mediator, and $Y(t, m)$ is the potential outcome variable. The average direct effect (ADE) is defined similarly as,

$$\zeta(t) \;=\; E\{Y(t_1, M(t)) - Y(t_0, M(t))\},$$

which represents the expected difference in the potential outcome when the treatment is changed but the mediator is held constant at the value that would realize if the treatment equals t. The two quantities on average add up to the total effect of the treatment on the outcome, τ. See the references for more details.

When both the mediator model ('model.m') and outcome model ('model.y') are linear, the results will be identical to the usual LSEM method by Baron and Kenny (1986). The function can, however, accommodate other data types including binary, ordered and count outcomes and mediators as well as censored outcomes. Variables can also be modeled nonparametrically, semiparametrically, or using quantile regressions.

The prior weights in the mediator and outcome models are taken as sampling weights and the estimated effects will be weighted averages when non-NULL weights are used in fitting 'model.m' and 'model.y'. This will be useful when data does not come from a simple random sample, for example.

As of version 3.0, the mediator model can be of either 'lm', 'glm', 'polr', 'gam', or 'rq' class, corresponding respectively to the linear regression models, generalized linear models, ordered response models, generalized additive models, or quantile regression models. Version 3.1 further added support for 'bayespolr' model for ordered response models. For binary response models, the 'mediator'

must be a numeric variable with values 0 or 1 as opposed to a factor. Quasi-likelihood-based inferences are not allowed for the mediator model because the functional form must be exactly specified for the estimation algorithm to work. The 'binomial' family can only be used for binary response mediators and cannot be used for multiple-trial responses. This is due to conflicts between how the latter type of models are implemented in `glm` and how 'mediate' is currently written.

For the outcome model, the censored regression model fitted via package `VGAM` (of class 'vglm' with 'family@vfamily' equal to "tobit") can be used in addition to the models listed above for the mediator. The 'mediate' function is not compatible with censored regression models fitted via other packages. When the quantile regression is used for the outcome model ('rq'), the estimated quantities are quantile causal mediation effects, quantile direct effects and etc., instead of the average effects.

The quasi-Bayesian approximation (King et al. 2000) cannot be used if 'model.m' is of class 'rq' or 'gam', or if 'model.y' is of class 'gam', 'polr' or 'bayespolr'. In these cases, either error is returned or nonparametric bootstrap is forced. Users should note that use of the nonparametric bootstrap often requires significant computing time, especially when 'sims' is set to a large value.

The 'control' argument must be provided when 'gam' is used for the outcome model and user wants to allow ACME and ADE to vary as functions of t (i.e., to relax the "no interaction" assumption). Note that the outcome model must be fitted via package `mgcv` with appropriate formula using `s` constructs (see Imai et al. 2009 in the references). For other model types, the interaction can be allowed by including an interaction term between T and M in the linear predictor of the outcome model. As of version 3.0, the 'INT' argument is deprecated and the existence of the interaction term is automatically detected (except for 'gam' outcome models).

When the treatment variable is continuous, user must specify the values of t_1 and t_0 using the 'treat.value' and 'control.value' arguments, respectively. The value of t in the above expressions is set to t_0 for 'd0', 'z0', etc. and to t_1 for 'd1', 'z1', etc.

Value

`mediate` returns an object of class "`mediate`" (or "`mediate.order`" if the outcome model used is 'polr' or 'bayespolr'), a list that contains the components listed below. Some of these elements are not available if 'long' is set to 'FALSE' by the user.

The function `summary` (i.e., `summary.mediate` or `summary.mediate.order`) can be used to obtain a table of the results. The function `plot` (i.e., `plot.mediate` or `plot.mediate.order`) can be used to produce a plot of the estimated average causal mediation, average direct, and total effects along with their confidence intervals.

`d0, d1` point estimates for average causal mediation effects under the control and treatment conditions.

`d0.ci, d1.ci` confidence intervals for average causal mediation effects. The confidence level is set at the value specified in 'conf.level'.

`d0.sims, d1.sims` vectors of length 'sims' containing simulation draws of average causal mediation effects.

`z0, z1` point estimates for average direct effect under the control and treatment conditions.

`z0.ci, z1.ci` confidence intervals for average direct effects.

`z0.sims, z1.sims`
: vectors of length 'sims' containing simulation draws of average direct effects.

`n0, n1`
: the "proportions mediated", or the size of the average causal mediation effects relative to the total effect.

`n0.ci, n1.ci`
: confidence intervals for the proportions mediated.

`n0.sims, n1.sims`
: vectors of length 'sims' containing simulation draws of the proportions mediated.

`tau.coef`
: point estimate for total effect.

`tau.ci`
: confidence interval for total effect.

`tau.sims`
: a vector of length 'sims' containing simulation draws of the total effect.

`d.avg, z.avg, n.avg`
: simple averages of d0 and d1, z0 and z1, n0 and n1, respectively, which users may want to use as summary values when those quantities differ.

`d.avg.ci, z.avg.ci, n.avg.ci`
: confidence intervals for the above.

`d.avg.sims, z.avg.sims, n.avg.sims`
: vectors of length 'sims' containing simulation draws of d.avg, z.avg and n.avg, respectively.

`boot`
: logical, the 'boot' argument used.

`treat`
: a character string indicating the name of the 'treat' variable used.

`mediator`
: a character string indicating the name of the 'mediator' variable used.

`INT`
: a logical value indicating whether the model specification allows the effects to differ between the treatment and control conditions.

`conf.level`
: the confidence level used.

`model.y`
: the outcome model used.

`model.m`
: the mediator model used.

`control.value`
: value of the treatment variable used as the control condition.

`treat.value`
: value of the treatment variable used as the treatment condition.

`nobs`
: number of observations in the model frame for 'model.m' and 'model.y'. May differ from the numbers in the original models input to 'mediate' if 'dropobs' was 'TRUE'.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>, Luke Keele, Ohio State University, <keele.4@osu.edu>, Kosuke Imai, Princeton University, <kimai@princeton.edu>

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

medsens, plot.mediate, summary.mediate, mediations, vcovHC

Examples

```
# Examples with JOBS II Field Experiment

# **For illustration purposes a small number of simulations are used**

data(jobs)

######################################################
# Example 1: Linear Outcome and Mediator Models
######################################################
b <- lm(job_seek ~ treat + econ_hard + sex + age, data=jobs)
c <- lm(depress2 ~ treat + job_seek + econ_hard + sex + age, data=jobs)

# Estimation via quasi-Bayesian approximation
contcont <- mediate(b, c, sims=50, treat="treat", mediator="job_seek")
summary(contcont)
plot(contcont)

# Estimation via nonparametric bootstrap
contcont.boot <- mediate(b, c, boot=TRUE, sims=50, treat="treat", mediator="job_seek")
summary(contcont.boot)

# Allowing treatment-mediator interaction
d <- lm(depress2 ~ treat + job_seek + treat:job_seek + econ_hard + sex + age, data=jobs)
contcont.int <- mediate(b, d, sims=50, treat="treat", mediator="job_seek")
summary(contcont.int)

# Continuous treatment
jobs$treat_cont <- jobs$treat + rnorm(nrow(jobs))  # (hypothetical) continuous treatment
b.contT <- lm(job_seek ~ treat_cont + econ_hard + sex + age, data=jobs)
c.contT <- lm(depress2 ~ treat_cont + job_seek + econ_hard + sex + age, data=jobs)
contcont.cont <- mediate(b.contT, c.contT, sims=50,
                    treat="treat_cont", mediator="job_seek",
                    treat.value = 4, control.value = -2)
summary(contcont.cont)

######################################################
```

```
# Example 2: Binary Outcome and Ordered Mediator
######################################################
b.ord <- polr(job_disc ~ treat + econ_hard + sex + age, data=jobs,
            method="probit", Hess=TRUE)
d.bin <- glm(work1 ~ treat * job_disc + econ_hard + sex + age, data=jobs,
            family=binomial(link="probit"))
ordbin <- mediate(b.ord, d.bin, sims=50, treat="treat", mediator="job_disc")
summary(ordbin)

# Using heteroskedasticity-consistent standard errors
require(sandwich)
ordbin.rb <- mediate(b.ord, d.bin, sims=50, treat="treat", mediator="job_disc",
            robustSE=TRUE)
summary(ordbin.rb)

######################################################
# Example 3: Quantile Causal Mediation Effect
######################################################
require(quantreg)
c.quan <- rq(depress2 ~ treat + job_seek + econ_hard + sex + age, data=jobs,
            tau = 0.5)  # median
contquan <- mediate(b, c.quan, sims=50, treat="treat", mediator="job_seek")
summary(contquan)

######################################################
# Example 4: GAM Outcome
######################################################
require(mgcv)
c.gam <- gam(depress2 ~ treat + s(job_seek, bs="cr") +
            econ_hard + sex + age, data=jobs)
contgam <- mediate(b, c.gam, sims=10, treat="treat",
                mediator="job_seek", boot=TRUE)
summary(contgam)

# With interaction
d.gam <- gam(depress2 ~ treat + s(job_seek, by = treat) +
    s(job_seek, by = control) + econ_hard + sex + age, data=jobs)
contgam.int <- mediate(b, d.gam, sims=10, treat="treat", mediator="job_seek",
    control = "control", boot=TRUE)
summary(contgam.int)
```

`mediations` *Causal Mediation Analysis for Multiple Outcome/Treatment/Mediator Combinations*

Description

'mediations' can be used to process a set of outcome/treatment/mediator combinations through the `mediate` function to produce a series of causal mediation analysis results.

Usage

```
mediations(datasets, treatment, mediators, outcome,
        covariates = NULL, families = c("gaussian", "gaussian"),
      tau.m = 0.5, tau.y = 0.5, LowerY = NULL, UpperY = NULL, interaction = FALSE,
        conf.level = .95, sims = 500, boot = FALSE, weights=NULL, ...)
```

Arguments

datasets	a named list of data frames. Each data frame has a separate treatment variable. The names of each data frame must begin with the exact name of the treatment variable that is contained in that dataset (see example below).
treatment	a vector of character strings indicating the names of the treatment variables, with length equal to the length of 'datasets'. Each treatment variable must be included in the data frame listed in the same position of list 'datasets' and its name must match the first part of the corresponding data frame.
mediators	a vector of character strings indicating the names of the mediators contained within each data frame. All of the mediators will be used with each treatment variable and hence must be included in each data frame of 'datasets'.
outcome	a vector of character strings indicating the names of the outcome variables contained within each data frame. All of the outcomes will be used with each treatment variable and must be in each data frame.
covariates	a character string representing the set of pre-treatment covariate names (as they appear in the data frame) to be included in each model. The value must take the form of standard model formula, with each additive component separated by "+", etc. (see example below). All covariates must be in each data frame. Default is 'NULL'.
families	a vector of length two specifying the types of the mediator and outcome models. Currently only supports "gaussian" (for linear regression), "binomial" (for binary probit), "oprobit" (for ordered probit) and "quantile" (for quantile regression, see 'tau'). For the outcome the tobit model ("tobit") is also available in addition to the mediator model options.
tau.m	a numeric value specifying the quantile to be used for a quantile regression for the mediator model. Only relevant if the first element of 'families' is "quantile". See rq.
tau.y	a numeric value specifying the quantile to be used for a quantile regression for the outcome model. Only relevant if the second element of 'families' is "quantile". See rq.
LowerY	a numeric value indicating the lower bound for the tobit outcome model. See tobit.
UpperY	a numeric value indicating the upper bound for the tobit outcome model. See tobit.
interaction	a logical value indicating whether the treatment and mediator variables should be interacted. This will apply to applications of mediate to all the treatment/mediator/outcome combinations.

conf.level	confidence level used in each application of the mediate function.
sims	an integer indicating the desired number of simulations to run mediate for. This will apply to all applications of 'mediate' to all the treatment/mediator/outcome combinations.
boot	a logical value, indicating whether or not nonparametric bootstrap should be used in each mediate application.
weights	a single valued vector of a character string indicating a weight variable to be used in all model fitting.
...	other arguments passed to mediate, such as 'robustSE', 'dropobs', etc.

Details

This function processes multiple treatment/mediators/outcome variable combinations to produce a collected set of output ready for analysis or graphing. In principle, this is a function designed to facilitate running causal mediation analyses on multiple models that share the same basic specification (i.e. the types of parametric models and the set of pre-treatment covariates) except the treatment, mediator and outcome variables can differ across specifications. The function works by looping over a set of data frames that are pre-loaded into the workspace. Each one of these data frames has a specific treatment variable that is used for analysis with that data frame. Then the code runs causal mediation analysis via mediate on every combination of the treatment, mediator, and outcomes specified in these arguments. This allows the users to explore whether different mediators transmit the effect of the treatment variable on a variety of outcome variables. A single set of pre-treatment control variables can be specified in 'covariates', which will be used throughout.

The 'mediations' function can be used with either multiple mediators and a single outcome, a single mediator and multiple outcomes, or multiple mediators and outcomes. For example, with three different treatments, user will create three different data frames, each containing a treatment variable. In addition, if there are also four different mediators, each of these will be contained in each data frame, along with the outcome variable. The function will estimate all of the combinations of treatment variables and mediators instead of separate lines of code being written for each one.

Individual elements of the output list (see "Value") may be passed through summary and plot for tabular and graphical summaries of the results. Alternatively, the entire output may be directly passed to summary or plot for all results to be inspected.

The default value of 'covariates' is 'NULL' and no covariate will be included in either mediator or outcome models without a custom value. It should be noted that users typically should have pre-treatment covariates to make the sequential ignorability assumption more plausible.

There are several limitations to the code. First, it works only with a subset of the model types that will be accommodated if 'mediate' is used individually (see the 'families' argument above for details). Second, one cannot specify separate sets of covariates for different treatment/mediator/outcome combinations. Users should use 'mediate' separately for individual models if more flexibility is required in their specific applications.

Value

An object of class "mediations" (or "mediations.order" if the outcome model is ordered probit), a list of "mediate" ("mediate.order") objects produced by applications of mediate for the specified treatment/mediator/outcome combinations. The elements are named based on the names of the outcome, treatment, and mediator variables, each separated by a "." (see example below).

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>.

See Also

mediate, summary.mediations, plot.mediations, rq, tobit.

Examples

```
## Not run:
# Hypothetical example

datasets <- list(T1 = T1, T2 = T2)
   # List of data frames corresponding to the two different treatment variables "T1vsCont" and "T2vsCont".
    # Each data set has its respective treatment variable.

mediators <- c("M1", "M2")
    # Vector of mediator names, all included in each data frame.

outcome <- c("Ycont1","Ycont2")
    # Vector of outcome variable names, again all included in each data frame.

treatment <- c("T1vsCont", "T2vsCont")
   # Vector of treatment variables names; must begin with identical strings with dataset names in 'datasets'.

covariates <- c("X1 + X2")
   # Set of covariates (in each data set), entered using the standard model formula format.

x <- mediations(datasets, treatment, mediators, outcome, covariates,
    families=c("gaussian","gaussian"), interaction=FALSE,
    conf.level=.90, sims=50)
   # Runs 'mediate' iteratively for each variable combinations, with 'lm' on both mediator and outcome model.

summary(x)  # tabular summary of results for all model combinations
plot(x)  # graphical summary of results for all model combinations at once

plot(x$Ycont1.T1vsCont.M1)
   # Individual 'mediate' outputs are stored as list elements and can be accessed using the usual "$" operator.

## End(Not run)
```

---

medsens *Sensitivity Analysis for Causal Mediation Effects*

Description

'medsens' is used to perform sensitivity analysis on the average causal mediation effects and direct effects for violations of the sequential ignorability assumption. The function takes output from 'mediate' and calculates the true average causal mediation effects and direct effects for different values of the sensitivity parameter representing the degree of the sequantial ignorability violation.

Usage

```
medsens(x, rho.by = 0.1, sims = 1000, eps = sqrt(.Machine$double.eps),
        effect.type = c("indirect", "direct", "both"))
```

Arguments

x	an object of class 'mediate', typically an output from the mediate function.
rho.by	a numeric value between 0 and 1 indicating the increment for the sensitivity parameter, rho.
sims	the number of Monte Carlo draws for the calculation of confidence intervals. Only used in cases where either the mediator or outcome variable is binary.
eps	convergence tolerance parameter for the iterative FGLS. Only used when both the mediator and outcome models are linear.
effect.type	a character string indicating which effect(s) to be analyzed. Default is "indirect".

Details

This is the workhorse function for sensitivity analyses for average causal mediation effects. The sensitivity analysis can be used to assess the robustness of the findings from mediate to the violation of sequantial ignorability, the crucial identification assumption necessary for the estimates to be valid. The analysis proceeds by quantifying the degree of sequantial ignorability violation as the correlation between the error terms of the mediator and outcome models, and then calculating the true values of the average causal mediation effect for given values of this sensitivity parameter, rho. The original findings are deemed sensitive if the true effects are found to vary widely as function of rho.

The sensitivity analysis is only implemented for the following three model combinations: linear mediator and outcome models (both of class 'lm'), binary probit mediator (fitted via 'glm' with family "binomial" and link "probit") and linear outcome models, and linear mediator and binary probit outcome models. An error is returned if the 'mediate' object in 'x' is based on other model combinations. As of version 3.0, the sensitivity analysis can also be conducted with respect to the average direct effect by setting 'effect.type' to "direct" (or "both" if results for the average causal mediation effect are also desired).

Users should note that computation can take significant time for medsens. Setting 'rho.by' to a larger number significantly decreases computational time, as does decreasing 'eps' (for the linear-linear case) or the number of simulations 'sims' (for the binary-linear and linear-binary cases).

Value

medsens returns an object of class "medsens", a list containing the following elements. Some of these elements are not available depending on the 'effect.type' argument specified by the user. The output can then be passed to the summary (i.e., summary.medsens) and plot (i.e., plot.medsens) functions to produce tabular and graphical summaries of the results.

d0, d1
: vectors of point estimates for average causal mediation effects under the control and treatment conditions for each value of sensitivity parameter rho.

upper.d0, lower.d0, upper.d1, lower.d1
: vectors of upper and lower confidence limits for average causal mediation effect under the control and treatment conditions for each value of rho.

z0, z1
: vectors of point estimates for average direct effect under the control and treatment conditions for each value of sensitivity parameter rho.

upper.z0, lower.z0, upper.z1, lower.z1
: vectors of upper and lower confidence limits for average direct effect under the control and treatment conditions for each value of rho.

tau
: a vector of point estimates for total effect for each value of rho. Only present when the outcome model is binary.

upper.tau, lower.tau
: vectors of upper and lower confidence limits for total effect. Only present when the outcome model is binary.

nu
: a vector of point estimates for the proportion mediated for each value of rho. Only present when the outcome model is binary.

upper.nu, lower.nu
: vectors of upper and lower confidence limits for the proportion mediated. Only present when the outcome model is binary.

rho
: a numeric vector containing the values of sensitivity parameter rho used.

rho.by
: a numeric value indicating the increment of rho used.

sims
: a numeric value indicating the number of Monte Carlo draws used.

err.cr.d, err.cr.z
: the values of rho with which the average causal mediation and direct effects are zero. Vectors of length two if 'INT' is 'TRUE'; numeric values otherwise.

ind.d0, ind.d1, ind.z0, ind.z1
: vectors of 0s/1s, indicating whether the confidence intervals of d0, d1, z0 and z1 do not cover zero for each value of rho.

R2star.prod
: a numeric vector containing the values of the products of the two "R square stars", representing the proportions of residual variance in the mediator and outcome explained by the hypothesized unobserved confounder. The values correspond to those of rho. See plot.medsens for details.

R2tilde.prod
: a numeric vector containing the values of the products of the two "R square tildes", representing the proportions of total variance in the mediator and outcome explained by the hypothesized unobserved confounder. The values correspond to those of rho. See plot.medsens for details.

`R2star.d.thresh, R2star.z.thresh`
: the values of the product of "R square stars" for which the average causal mediation and direct effects are zero, respectively.

`R2tilde.d.thresh, R2tilde.z.thresh`
: the values of the product of "R square tildes" for which the average causal mediation and direct effects are zero, respectively.

`r.square.y, r.square.m`
: the usual R square statistics for the outcome and mediator models.

`INT`
: a logical value indicating whether interaction between the treatment and mediator is allowed in the original mediate object.

`conf.level`
: the confidence level used.

`effect.type`
: the 'effect.type' argument used.

`type`
: a character string indicating the type of the mediator and outcome models used. Currently either "ct" (linear mediator and outcome models), 'bm' (binary mediator and linear outcome models) or 'bo' (linear mediator and binary outcome models).

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>, Jaquilyn Waddell-Boie, Princeton University, <jwaddell@princeton.edu>, Luke Keele, Ohio State University, <keele.4@osu.edu>, Kosuke Imai, Princeton University, <kimai@princeton.edu>

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

`mediate`, `summary.medsens`, `plot.medsens`.

Examples

```
# Examples with JOBS II Field Experiment

# **For illustration purposes a small number of simulations are used**

data(jobs)

# Fit parametric models
b <- lm(job_seek ~ treat + econ_hard + sex + age, data=jobs)
```

```
c <- lm(depress2 ~ treat + job_seek + econ_hard + sex + age, data=jobs)

# Pass model objects through mediate function
med.cont <- mediate(b, c, treat="treat", mediator="job_seek", sims=50)

# Pass mediate output through medsens function
sens.cont <- medsens(med.cont, rho.by=.1, eps=.01, effect.type="both")

# Use summary function to display results
summary(sens.cont)

# Plot true ACMEs and ADEs as functions of rho
par.orig <- par(mfrow = c(2,2))
plot(sens.cont, main="JOBS", ylim=c(-.2,.2))

# Plot true ACMEs and ADEs as functions of "R square tildes"
plot(sens.cont, sens.par="R2", r.type="total", sign.prod="positive")
par(par.orig)
```

`plot.mediate` *Plotting Indirect, Direct, and Total Effects from Mediation Analysis*

Description

Function to plot results from `mediate`. The vertical axis lists indirect, direct, and total effects and the horizontal axis indicates the respective magnitudes. Most standard options for plot function available.

Usage

```
## S3 method for class 'mediate'
plot(x, treatment = NULL,
    labels = c("ACME","Direct\nEffect","Total\nEffect"),
    xlim = NULL, ylim = NULL, xlab = "", ylab = "",
    main = NULL, lwd = 1.5, cex = .85,
    col = "black", ...)
```

Arguments

`x`	object of class `mediate` or `mediate.order` as produced by `mediate`.
`treatment`	a character string indicating the baseline treatment value of the estimated causal mediation effect and direct effect to plot. Can be either "control", "treated" or "both". If 'NULL' (default), both sets of estimates are plotted if and only if they differ.
`labels`	a vector of length 3 indicating the labels for the estimated effects.
`xlim`	range of the horizontal axis.
`ylim`	range of the vertical axis.

xlab	label of the horizontal axis.
ylab	label of the vertical axis.
main	main title.
lwd	width of the horizontal bars for confidence intervals.
cex	size of the dots for point estimates.
col	color of the dots and horizontal bars for the estimates.
...	additional parameters passed to 'plot'.

Value

mediate returns an object of class "mediate". The function summary is used to obtain a table of the results. The plot function plots these quantities.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>.

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

mediate, plot

---

plot.mediations *Plotting Indirect, Direct, and Total Effects from Multiple Mediation Analyses*

Description

Function to plot results from multiple causal mediation analyses conducted via the mediations funciton. Output is a series of plots generated via plot.mediate for each treatment/mediator/outcome combination specified in the input 'mediations' object.

Usage

```
## S3 method for class 'mediations'
plot(x, which = names(x),
            ask = prod(par("mfcol")) < length(which) && dev.interactive(), ...)
```

Arguments

x	output from the mediations function.
which	subset of names(x), indicating which model combinations to be plotted. Default is to plot all.
ask	logical. If 'TRUE', the user is asked for input before a new figure is plotted. Default is to ask only if the number of plots on current screen is fewer the number implied by 'which'.
...	arguments passed to the plot.mediate function for individual plots.

Value

mediations returns an object of class mediations. The function summary is used to obtain a table of the results. The plot function instead plots these quantities. All additional parameters desired for the plotting of an output from mediate can be passed through.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>.

See Also

mediations, plot.mediate, plot.

---

plot.medsens *Plotting Results from Sensitivity Analysis for Causal Mediation Effects*

Description

This function is used to plot results from the 'medsens' function. Causal average mediation effects (as well as average direct effects and proportions mediated for selected models) can be plotted against two alternative sensitivity parameters.

Usage

```
  ## S3 method for class 'medsens'
plot(x, sens.par = c("rho", "R2"),
       r.type = c("residual", "total"), sign.prod = c("positive", "negative"),
       pr.plot = FALSE, smooth.effect = FALSE, smooth.ci = FALSE,
       ask = prod(par("mfcol")) < nplots, levels = NULL,
       xlab = NULL, ylab = NULL, xlim = NULL, ylim = NULL,
       main = NULL, lwd = par("lwd"), ...)
```

Arguments

x	'medsens' object, typically output from medsens.
sens.par	a character string indicating the sensitivity parameter to be used. Default plots effects as functions of "rho". See Details.
r.type	type of the R square parameter to be used in "R2" plots. If "residual", effects are plotted against the proportions of the residual variances that are explained by the unobserved confounder. If "total", the proportions of the total variances are used as sensitivity parameters. Only relevant if 'sens.par' is "R2".
sign.prod	a value indicating the direction of hypothesized confounding in the sensitivity analysis. If "positive", the confounder is assumed to affect the mediator and outcome variable in the same direction; if "negative" the effects are assumed to be in opposite directions. Only relevant if sens.par is set to "R2".
pr.plot	a logical value. If 'TRUE', the "proportions mediated" will be plotted instead of the average causal mediation effects or direct effects. Currently only available if the object 'medsens' is based on the linear mediator and binary probit outcome models. Default is 'FALSE'.
smooth.effect	a logical value indicating whether the estimated mediation effects are smoothed via lowess before being plotted. Default is 'FALSE'.
smooth.ci	a logical value indicating whether the confidence bands are smoothed via lowess before being plotted. Default is 'FALSE'.
ask	a logical value. If 'TRUE', the user is asked for input before a new figure is plotted. Default is to ask only if the number of plots on current screen is fewer than necessary.
levels	vector of levels at which to draw contour lines. Only relevant if 'sens.par' is set to "R2". If 'NULL', default values in contour.default are used.
xlab	label for the x axis. Default labels are used if 'NULL'.
ylab	label for the y axis. Default labels are used if 'NULL'.
xlim	limits of the x axis. If 'NULL' default values are used.
ylim	limits of the y axis. If 'NULL' default values are used.
main	main title for the plot. If 'NULL', default titles are used.
lwd	width of the lines used in graphs.
...	additional arguments to be passed to plotting functions.

Details

The sensitivity analysis for causal mediation effects can be conducted in terms of two alternative sensitivity parameters, which both quantify the degree of violation of the sequential ignorability assumption. The "rho" parameter represents the correlation between the two error terms of the (latent) linear models for the mediator and outcome variables. A large value of rho indicates the existence of important common unobserved predictors for both the mediator and outcome and therefore a high degree of sequential ignorability violation, while a value close to zero implies there is no such confounders.

The resulting "rho" figures plot the estimated true values of ACME (or ADE, proportion mediated) against rho, along with the confidence intervals. When rho is zero, sequantial ignorability holds, so the estimated value at that point will be equal to the estimate returned by the `mediate`. The confidence level is determined by the 'conf.level' value of the original `mediate` object.

The "R2" parameters represent the proportions of the mediator and outcome variances that are explained by an unobserved pre-treatment confounder, thereby indicating the importance of such a confounder in each model. When 'r.type' is "residual", the R2 parameters represent the proportions of the residual variances of the mediator and outcome models that become explained by the inclusion of the hypothetical pre-treatment confounder. These are denoted as "R square stars" in Imai, Keele and Yamamoto (2010) and can also be specified as "star" or using a numeric value 1 in `medsens.plot`. When 'r.type' is "total", the R2s represent the total mediator and outcome variances the unobserved confounder would explain. This option can also be specified using "tilde" or a numeric value 2.

For both types of the "R2" parameters, 'sign.prod' indicates the hypothesized direction in which the unobserved confounder affects the mediator and outcome. (The name derives from the fact that this direction is mathematically represented by the sign of the product of two regression coefficients.) If "positive" (or a numeric value 1) is given, the confounder is assumed to affect the mediator and outcome in the same direction. If "negative" (or a numeric value -1), the effect is assumed to be in opposite directions.

The resulting contours in the "R2" plots represent the values of the ACME (or ADE) for different combinations of the mediator R2 and outcome R2 values. When both values are zero (the lower-left corner of the plot), the unobserved pre-treatment confounder has no effect on either mediator or outcome and therefore sequantial ignorability is satisfied.

Warning

The 'smooth.effect' and 'smooth.ci' options should be used with caution since the smoothing could affect substantive implications of the graphical analysis in a significant way.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>, Jaquilyn Waddell-Boie, Princeton University, <jwaddell@princeton.edu>, Luke Keele, Ohio State University, <keele.4@osu.edu>, Kosuke Imai, Princeton University, <kimai@princeton.edu>

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

medsens, plot, contour.

---

summary.mediate *Summarizing Output from Mediation Analysis*

Description

Function to report results from mediation analysis. Reported categories are mediation effect, direct effect, total effect, and proportion of total effect mediated. All quantities reported with confidence intervals. If the treatment-mediator interaction is allowed in the mediation analysis, effects are reported separately for the treatment and control conditions as well as the simple averages of these effects are displayed at the bottom of the summary table.

Usage

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

  ## S3 method for class 'summary.mediate'
print(x, ...)
```

Arguments

object	output from mediate function.
x	output from summary.mediate function.
...	additional arguments affecting the summary produced.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>, Luke Keele, Ohio State University, <keele.4@osu.edu>, Kosuke Imai, Princeton University, <kimai@princeton.edu>

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

mediate, plot.mediate, summary.

---

summary.mediations *Summarizing Output from Multiple Mediation Analyses*

Description

The 'summary.mediations' function produces a summary of results from multiple causal analyses conducted via mediations. Output is a series of summary.mediate outputs for all the treatment/mediator/outcome combinations used in the input 'mediations' object.

Usage

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

  ## S3 method for class 'summary.mediations'
print(x, ...)
```

Arguments

object	output from mediations function.
x	output from summary.mediations function.
...	additional arguments affecting the summary produced.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>.

See Also

mediations, summary.mediate, summary.

summary.medsens *Summarizing Results from Sensitivity Analysis for Causal Mediation Effects*

Description

Functions to report results from the sensitivity analysis for causal mediation effects via medsens in a tabular form.

Usage

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

  ## S3 method for class 'summary.medsens'
print(x, ...)
```

Arguments

object	output from medsens function.
x	output from summary.medsens function.
...	additional arguments affecting the summary produced.

Author(s)

Dustin Tingley, Harvard University, <dtingley@gov.harvard.edu>, Teppei Yamamoto, Princeton University, <tyamamot@princeton.edu>, Jaquilyn Waddell-Boie, Princeton University, <jwaddell@princeton.edu>, Luke Keele, Ohio State University, <keele.4@osu.edu>, Kosuke Imai, Princeton University, <kimai@princeton.edu>

References

Imai, K., Keele, L. and Tingley, D. (2010) A General Approach to Causal Mediation Analysis, Psychological Methods, Vol. 15, No. 4 (December), pp. 309-334.

Imai, K., Keele, L. and Yamamoto, T. (2010) Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects, Statistical Science, Vol. 25, No. 1 (February), pp. 51-71.

Imai, K., Keele, L., Tingley, D. and Yamamoto, T. (2009) "Causal Mediation Analysis Using R" in Advances in Social Science Research Using R, ed. H. D. Vinod New York: Springer.

See Also

medsens, summary.

Index