mediation: R Package for Causal Mediation Analysis

Dustin Tingley Harvard
Teppei Yamamoto MIT
Kentaro Hirose Princeton
Luke Keele Penn State
Kosuke Imai Princeton

Abstract

In this paper, we describe the R package **mediation** for conducting causal mediation analysis in applied empirical research. In many scientific disciplines, the goal of researchers is not only estimating causal effects of a treatment but also understanding the process in which the treatment causally affects the outcome. Causal mediation analysis is frequently used to assess potential causal mechanisms. The **mediation** package implements a comprehensive suite of statistical tools for conducting such an analysis. The package is organized into two distinct approaches. Using the model-based approach, researchers can estimate causal mediation effects and conduct sensitivity analysis under the standard research design. Furthermore, the design-based approach provides several analysis tools that are applicable under different experimental designs. This approach requires weaker assumptions than the model-based approach. We also implement a statistical method for dealing with multiple (causally dependent) mediators, which are often encountered in practice. Finally, the package also offers a methodology for assessing causal mediation in the presence of treatment noncompliance, a common problem in randomized trials.

Keywords: causal mechanisms, mediation analysis, **mediation**, R.

1. Introduction

Scholars across a wide range of disciplines are increasingly interested in identifying causal mechanisms, going beyond the estimation of causal effects. Once they ascertain that certain variables causally affect the outcome, the next natural step is to understand how these variables exert their influence. The standard procedure for analyzing causal mechanisms in applied research is called *mediation analysis*, where a set of linear regression models are fitted and then the estimates of "mediation effects" are computed from the fitted models (e.g., Haavelmo 1943; Baron and Kenny 1986; Shadish, Cook, and Campbell 2001; MacKinnon 2008). In recent years, however, causal mechanisms have been studied within the modern framework of causal inference with an emphasis on the assumptions required for identification. This approach has highlighted limitations of earlier methods and pointed the way towards a more flexible estimation strategy. In addition, new research designs have been proposed for identifying causal mechanisms.

In this paper, we introduce a full featured R package, **mediation** (Tingley, Yamamoto, Hirose, Keele, and Imai 2013), for studying causal mechanisms. The **mediation** package allows users to (1) investigate the role of causal mechanisms using different types of data and statistical models, (2) explore how results change as identification assumptions are relaxed, and (3) calculate quantities of interest under alternative research designs. We focus on the demonstration of the functionalities available through the **mediation** package. The statistical theory

that underlies the procedures implemented in the **mediation** package is presented elsewhere along with various empirical examples (Imai, Keele, and Yamamoto 2010c; Imai, Keele, Tingley, and Yamamoto 2011; Imai, Keele, and Tingley 2010a; Imai, Tingley, and Yamamoto 2013; Yamamoto 2013).

The **mediation** package is freely available for download via the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=mediation` and runs on a variety of computing platforms (R Core Team 2014). In addition, a Stata (StataCorp. 2013) version of the package is available but has a more limited functionality (Hicks and Tingley 2011). The first version of the **mediation** package appeared at CRAN in 2009, and Imai, Keele, Tingley, and Yamamoto (2010b) discuss an earlier version of the package. Since then, however, we have dramatically improved the package with a significant number of new functionalities and improvements. The current paper thus provides an up-to-date description of the analyses that can be conducted via the **mediation** package. To install the **mediation** package, use the following standard syntax for installing an R package,

```
R> install.packages("mediation")
```

where users may be prompted to select a CRAN mirror from which the package will be downloaded. This step needs to be done only once (unless one wishes to update the **mediation** package to the new version).

In the next section, we present an overview of the **mediation** package. We then describe the functionalities of the package for the model-based causal mediation analysis (Section 3), multilevel mediation analysis (Section 4), the design-based causal mediation analysis (Section 5), the analysis of causally dependent multiple mediators (Section 6), and causal mediation analysis with treatment noncompliance (Section 7). Finally, Section 8 concludes.

2. Overview of the mediation package

The **mediation** package consists of several main functions as well as various methods for summarizing output from these functions (e.g., `plot` and `summary`). The package requires little programming knowledge on the user's side. Figure 1 illustrates the core structure of the **mediation** package, which distinguishes between model-based and design-based inference. Model-based inference has been standard practice in the mediation analysis to date. In the experimental setting, the treatment variable is randomized and the mediating and outcome variables are observed without any intervention by researchers. Imai et al. (2010a) show that a range of parametric and semi-parametric models may then be used to estimate the average causal mediation effect, defined below, and other quantities of interest. This modeling approach relies on the sequential ignorability assumption for point identification, which as Imai et al. (2010a) show, provides a general purpose algorithm for estimating quantities of interest. In contrast, design-based inference primarily employs the features of the experimental design and does not require the sequential ignorability assumption. The formal identification properties of these designs are studied by Imai et al. (2013) and the examples from experimental and observational studies are contained in Imai et al. (2011, 2013). We refer readers to these papers for the details about the statistical methods implemented via the **mediation** package.

Before describing the functions available in **mediation**, we briefly define the quantities of interest that our software is designed to estimate. Here, we use the potential outcomes framework

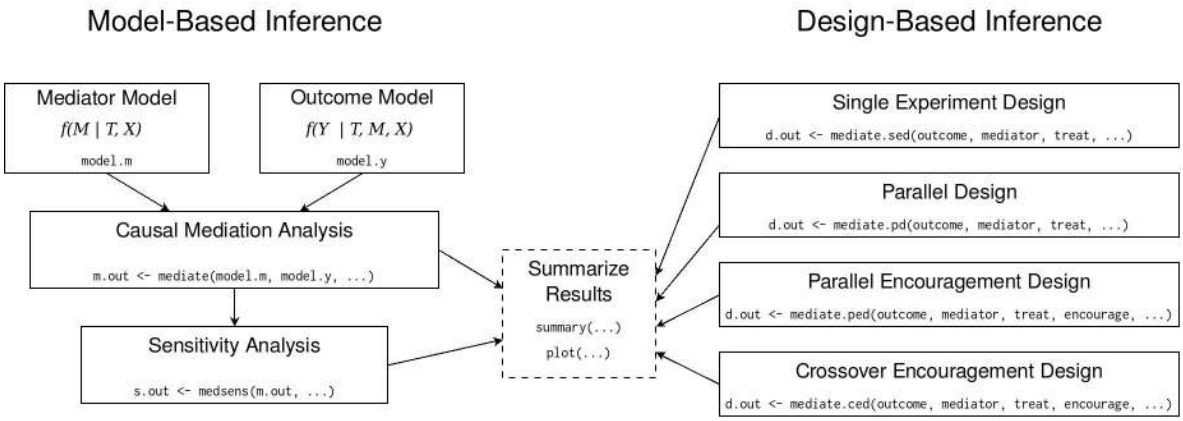

Figure 1: Core structure of the **mediation** package as of version 4.0.

to define these quantities. Let $M_i(t)$ denote the potential value of a mediator of interest for unit i under the treatment status $T_i = t$. Let $Y_i(t, m)$ denote the potential outcome that would result if the treatment and mediating variables equal t and m, respectively. Consider a standard experimental design where only the treatment variable is randomized. We observe only one of the potential outcomes, and the observed outcome, Y_i, equals $Y_i(T_i, M_i(T_i))$ where $M_i(T_i)$ represents the observed value of the mediator M_i. With this notation, the total unit treatment effect can be written as,

$$\tau_i \equiv Y_i(1, M_i(1)) - Y_i(0, M_i(0)). \tag{1}$$

We can decompose this total effect into the two components. First, the *causal mediation effects* are represented by (Robins and Greenland 1992; Pearl 2001),

$$\delta_i(t) \equiv Y_i(t, M_i(1)) - Y_i(t, M_i(0)), \tag{2}$$

for each treatment status $t = 0, 1$. All other causal mechanisms can be represented by the *direct effects* of the treatment as,

$$\zeta_i(t) \equiv Y_i(1, M_i(t)) - Y_i(0, M_i(t)), \tag{3}$$

for each unit i and each treatment status $t = 0, 1$. Together, we see that they sum up to the total effect,

$$\tau_i = \delta_i(t) + \zeta_i(1 - t) \tag{4}$$

for $t = 0, 1$. The case of multiple candidate mediating variables requires additional notation and is discussed in Section 6. The *average causal mediation effects* (ACME) $\bar{\delta}(t)$ and the average direct effects (ADE) $\bar{\zeta}(t)$, represent the population averages of these causal mediation and direct effects.

Identification of the ACME requires an additional assumption beyond the strong ignorability of the treatment, which is sufficient for identifying the average total effect of the treatment. Let X_i be a vector of the observed pre-treatment confounders for unit i. The key identifying assumption is called sequential ignorability and can be written as,

Assumption 1 (Sequential Ignorability; Imai et al. 2010c)

$$\{Y_i(t', m), M_i(t)\} \perp\!\!\!\perp T_i \mid X_i = x, \tag{5}$$

$$Y_i(t', m) \perp\!\!\!\perp M_i(t) \mid T_i = t, X_i = x, \tag{6}$$

where $0 < \mathsf{P}(T_i = t \mid X_i = x)$ *and* $0 < p(M_i = m \mid T_i = t, X_i = x)$ *for* $t = 0, 1$*, and all* x *and* m *in the support of* X_i *and* M_i*, respectively.*

Equation 5 is the standard strong ignorability of the treatment assignment and is satisfied, for example, if the treatment is randomized (possibly conditional on X_i). However, Equation 6 requires that the mediator is also ignorable given the observed treatment and pre-treatment confounders. This additional assumption is quite strong because it excludes the existence of (measured or unmeasured) post-treatment confounders as well as that of unmeasured pre-treatment confounders. This assumption, therefore, rules out the possibility of multiple mediators that are causally related to each other (see Section 6 for the method that is designed to deal with such a scenario).

3. Model-based causal mediation analysis

In this section, we discuss the functionalities of the **mediation** package for model-based causal mediation analysis under the assumption of sequential ignorability. Many of these functionalities are described in detail in Imai et al. (2010b), but the current version of the package accommodates a larger class of statistical models.

The model-based causal mediation analysis proceeds in two steps. First, the researcher specifies two statistical models, the mediator model for the conditional distribution of the mediator M_i given the treatment T_i and a set of the observed pre-treatment covariates X_i and the outcome model for the conditional distribution of the outcome Y_i given T_i, M_i, and X_i. These models are fitted separately and then their fitted objects comprise the main inputs to the `mediate` function, which computes the estimated ACME and other quantities of interest under these models and the sequential ignorability assumption. Since the sequential ignorability assumption is untestable, we recommend that the researchers conduct a sensitivity analysis via the `medsens` function, which can be applied to certain statistical models. We now illustrate these functionalities with an empirical example.

	Outcome model types						
Mediator model types	Linear	GLM	Ordered	Censored	Quantile	GAM	Survival
Linear (`lm`/`lmer`)	✓	✓	✓*	✓	✓	✓*	✓
GLM (`glm`/`bayesglm`/ `glmer`)	✓	✓	✓*	✓	✓	✓*	✓
Ordered (`polr`/`bayespolr`)	✓	✓	✓*	✓	✓	✓*	✓
Censored (tobit via `vglm`)	–	–	–	–	–	–	–
Quantile (`rq`)	✓*	✓*	✓*	✓*	✓*	✓*	✓
GAM (`gam`)	✓*	✓*	✓*	✓*	✓*	✓*	✓*
Survival (`survreg`)	✓	✓	✓*	✓	✓	✓*	✓

Table 1: Types of statistical models that can be used with the `mediate` function. Asterisks, *, indicate the model combinations that can only be estimated using the nonparametric bootstrap (i.e., with the argument `boot = TRUE` for the `mediate` function).

3.1. Estimation of the average causal mediation effects

The `mediate` function takes various standard model objects (such as obtained with `lm` and `glm`), which correspond to mediator and outcome models, and returns the estimates of the average causal mediation effects along with other causal quantities of interest. The output of the `mediate` function can be passed to the `plot` and `summary` functions in order to obtain graphical and numerical summaries, respectively. The `mediate` function automatically detects the type of models used for the mediator and outcome models and calculates the estimates of the ACME and other quantities of interest via the general algorithms described in Imai et al. (2010a). Our estimation strategy overcomes the limitation of the standard methods based on the product or difference of coefficients, which are only appropriate for the analysis of causal mediation effects when both the mediator and outcome models are linear regressions where T_i and M_i enter the models additively (e.g., without interaction). In contrast, the algorithms used in the **mediation** package nest this as a special case and accommodate a greater range of statistical models as shown in Table 1.

We now illustrate the use of the `mediate` function with an empirical example based on the `framing` data of Brader, Valentino, and Suhat (2008). This data set is a part of the **mediation** package and can be loaded via the following syntax,

```
R> library("mediation")
R> set.seed(2014)
R> data("framing", package = "mediation")
```

A brief description of each variable in the data can be obtained through a help file,

```
R> ?framing
```

Brader et al. (2008) conducted a randomized experiment where subjects are exposed to different media stories about immigration and the authors investigated how their framing influences attitudes and political behavior regarding immigration policy. They posit anxiety as the mediating variable for the causal effect of framing on public opinion. We first fit the mediator model where the measure of anxiety (`emo`) is modeled as a function of the framing treatment

(treat) and pre-treatment covariates (age, educ, gender, and income). Next, we model the outcome variable, which is a binary variable indicating whether or not the participant agreed to send a letter about immigration policy to his or her member of Congress (cong_mesg). The explanatory variables of the outcome model include the mediator, treatment status, and the same set of pre-treatment variables as those used in the mediator model.[1] In this example, the treatment is expected to increase the level of respondents' emotional response, which in turn is hypothesized to make subjects more likely to send a letter to his or her member of Congress. We use the linear regression fit with least squares and the probit regression for the mediator and outcome models, respectively.

```
R> med.fit <- lm(emo ~ treat + age + educ + gender + income, data = framing)
R> out.fit <- glm(cong_mesg ~ emo + treat + age + educ + gender + income,
+                 data = framing, family = binomial("probit"))
```

We now use the mediate function to estimate the ACME and ADE. As the inputs to this function, we must specify the model fits (in this case med.fit and out.fit) as well as the names of the treatment and mediating variables, which are represented as the arguments treat and mediator, respectively. Here and throughout the rest of this paper, we use a small number of simulations (sims = 100) for the purpose of illustration to calculate the uncertainty estimates, but one may wish to use the default (1000) or even larger number if the estimates vary too much from one simulation to another. The default simulation type is the quasi-Bayesian Monte Carlo method based on normal approximation (Imai et al. 2010a). We use White's heteroskedasticity-consistent estimator for the covariance matrix from the **sandwich** package (vcovHC; Zeileis 2006) by setting the robustSE argument to TRUE. This argument can be omitted if standard uncertainty estimates are desired. Finally, like most functions in R, the results of the mediate function can be summarized numerically by the summary function, which calculates point estimates, confidence intervals, and the p-values, for the average direct, indirect, and total effects.[2] The syntax is now given as,

```
R> med.out <- mediate(med.fit, out.fit, treat = "treat", mediator = "emo",
+                     robustSE = TRUE, sims = 100)
R> summary(med.out)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

                    Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)        0.0791       0.0351       0.1501    0.00
ACME (treated)        0.0804       0.0367       0.1557    0.00
ADE (control)         0.0206      -0.0976       0.1158    0.70
ADE (treated)         0.0218      -0.1053       0.1226    0.70
```

[1] Using different sets of pre-treatment covariates for the mediator and outcome models may be justified depending on the causal relationships assumed for those covariates. See Pearl (2014) and Imai, Keele, Tingley, and Yamamoto (2014).

[2] Note that the results will be slightly different in each run of mediate because of Monte Carlo errors, especially when sims is set to a small number. If exact reproduction of results is desired, users can set a specific randomness seed (set.seed) before calling the mediate function.

```
Total Effect               0.1009      -0.0497       0.2339    0.14
Prop. Mediated (control)   0.6946      -6.3109       3.6793    0.14
Prop. Mediated (treated)   0.7118      -5.7936       3.4965    0.14
ACME (average)             0.0798       0.0359       0.1537    0.00
ADE (average)              0.0212      -0.1014       0.1192    0.70
Prop. Mediated (average)   0.7032      -6.0523       3.5879    0.14

Sample Size Used: 265


Simulations: 100
```

One new feature in the tabular output from the `mediate` functions is the addition of p-values for the various estimates. In this example, the estimated ACMEs are statistically significantly different from zero but the estimated average direct and total effects are not. The results suggest that the treatment in the framing experiment may have increased emotional response, which in turn made subjects more likely to send a message to his or her member of Congress. Here, since the outcome is binary all estimated effects are expressed as the increase in probability that the subject sent a message to his or her Congress person.

In addition, we can use the nonparametric bootstrap rather than the quasi-Bayesian Monte Carlo simulation for variance estimation via the `boot = TRUE` argument,

```
R> med.out <- mediate(med.fit, out.fit, boot = TRUE, treat = "treat",
+                     mediator = "emo", sims = 100)
R> summary(med.out)

Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method

                         Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)             0.0832       0.0426       0.1332    0.00
ACME (treated)             0.0844       0.0425       0.1333    0.00
ADE (control)              0.0114      -0.1158       0.1277    0.84
ADE (treated)              0.0125      -0.1274       0.1360    0.84
Total Effect               0.0958      -0.0477       0.2171    0.24
Prop. Mediated (control)   0.8691      -3.4279       6.2842    0.24
Prop. Mediated (treated)   0.8811      -2.9262       5.9626    0.24
ACME (average)             0.0838       0.0434       0.1319    0.00
ADE (average)              0.0120      -0.1210       0.1318    0.84
Prop. Mediated (average)   0.8751      -3.1770       6.1234    0.24

Sample Size Used: 265


Simulations: 100
```

The output now indicates that the bootstrap is used for inferences. As expected, the results are largely the same. In general, as long as computing power is not an issue, analysts should estimate confidence intervals via the bootstrap with more than 1000 resamples, which is the default number of simulations.

Two types of methods for calculating bootstrap-based confidence intervals are available via the `boot.ci.type` argument. The basic percentile intervals are calculated by default or setting the argument to `"perc"`. The bias-corrected and accelerated (BCa) intervals are computed if the argument is set to `"bca"` (see DiCiccio and Efron 1996, for the definition of the method). The latter has better asymptotic properties and is often recommended for the estimation of mediation effects (Preacher and Hayes 2008).

As an alternative to the numerical summary, using the output from the `mediate` function as the input to the `plot` command provides a graphical summary of the three parameters (indirect, direct, and total effects) along with their confidence intervals. Figure 2 shows the result of plotting the `med.out` object.[3]

Treatment and mediator interaction

It is possible that the ACME takes different values depending on the baseline treatment status. In such a situation, the researcher can add an interaction term between the treatment and mediator to the outcome model. Then, the `mediate` function automatically detects the change in the specification and explicitly estimates the ACME conditional on treatment status.[4] In the output given below, the estimated ACME now varies with treatment status.

```
R> med.fit <- lm(emo ~ treat + age + educ + gender + income, data=framing)
R> out.fit <- glm(cong_mesg ~ emo * treat + age + educ + gender + income,
+                 data = framing, family = binomial("probit"))
R> med.out <- mediate(med.fit, out.fit, treat = "treat", mediator = "emo",
+                     robustSE = TRUE, sims = 100)
R> summary(med.out)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

                         Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)            0.07942      0.02497      0.14275    0.00
ACME (treated)            0.10362      0.03558      0.17073    0.00
ADE (control)             0.00319     -0.10976      0.13230    0.98
ADE (treated)             0.02739     -0.11584      0.16657    0.68
Total Effect              0.10682     -0.05053      0.24410    0.20
Prop. Mediated (control)  0.65447     -2.16982      3.57927    0.20
Prop. Mediated (treated)  0.80207     -2.28937      3.64659    0.20
```

[3] Users may make further modifications to the plot via standard `plot` options, including changes to the labels.

[4] When the outcome model is nonlinear, the ACME and direct effect estimates will differ between the treatment and control conditions even when the model does not include an interaction term. The `summary` output in such cases includes average values of these two estimates to ease interpretation of the results.

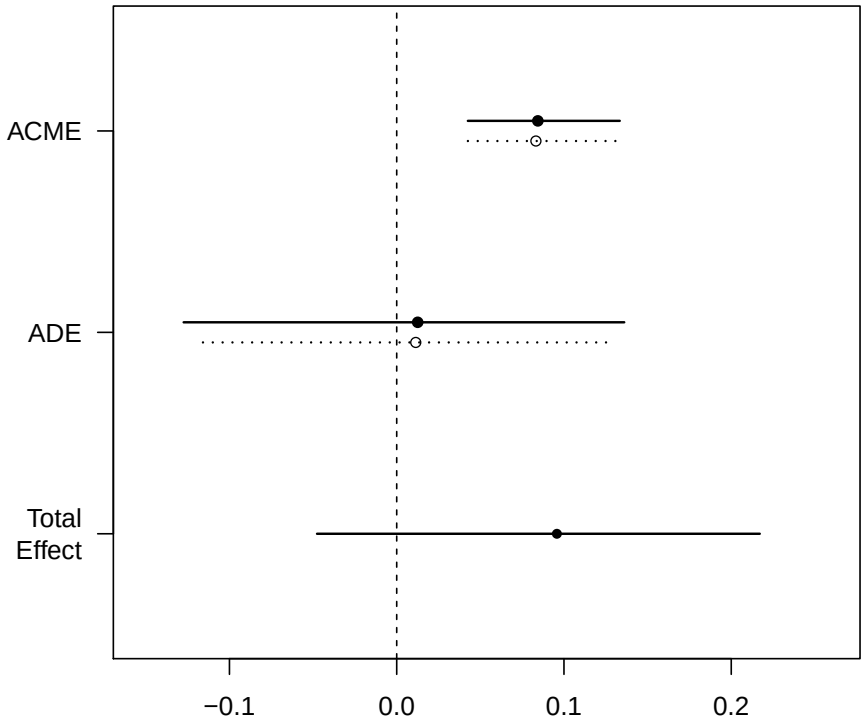

Figure 2: Graphical display of results from the `mediate` function.

```
ACME (average)            0.09152      0.03203      0.14967    0.00
ADE (average)             0.01529     -0.11744      0.14746    0.90
Prop. Mediated (average)  0.72827     -2.15158      3.54922    0.20

Sample Size Used: 265


Simulations: 100
```

The statistical significance of the treatment-mediator interaction can be tested via the `test.TMint` function in the following manner.

```
R> test.TMint(med.out, conf.level = .95)

        Test of ACME(1) - ACME(0) = 0

data:  estimates from med.out
```

```
ACME(1) - ACME(0) = 0.0242, p-value = 0.3
alternative hypothesis: true ACME(1) - ACME(0) is not equal to 0
95 percent confidence interval:
 -0.01795541  0.06809042
```

The `mediate` function's output contains a range of additional quantities that users might find helpful. Each is stored as part of the model's output. This includes vectors of the simulation output for all quantities of interests (see `?mediate` for details), which can be used for a variety of tasks, such as more intensive plotting.

Missing data

Our simulation-based approach to the estimation of mediation effects enables users to deal with missing data via standard multiple imputation procedures in a straightforward fashion. The **mediation** package includes a pair of utility functions – `mediations` and `amelidiate` – to facilitate such analysis. First, users simulate multiple data sets using their preferred imputation software. Next, run `mediate` on each data set by simply passing the data sets through `mediations`. Next, pass the output of `mediations` to the `amelidiate` function, which combines the components of the output from `mediations` into a format that can be analyzed with the standard `summary` and `plot` commands.[5] Alternatively, users can manually run `mediate` on their imputed data sets and simply stack the vectors of quantities they are interested in, and use basic functions like `quantile` to calculate confidence intervals.

3.2. Moderated mediation

One new important feature of the `mediate` function is the ability to study moderated mediation. Often analysts hypothesize that the magnitude of the ACME depends on (or is moderated by) a pre-treatment covariate. Such a pre-treatment covariate is called a moderator. In the framing example, the ACME may be much stronger among older respondents than younger ones. In other words, the ACME may be moderated by age.

There are two alternative routes to the analysis of moderated mediation with the **mediation** package. The first method involves alteration of both the statistical models as well as the syntax for the `mediate` function. First, the mediator and outcome models should contain the moderator and its interaction terms with respect to the treatment and mediating variables that are theoretically justified. For example, we may modify the previous models as follows,

```
R> med.fit <- lm(emo ~ treat * age + educ + gender + income, data=framing)
R> out.fit <- glm(cong_mesg ~ emo + treat * age + emo * age + educ + gender
+                 + income, data = framing, family = binomial("probit"))
```

Once the two models are fitted, the researcher must specify the levels of the moderator at which effects will be calculated by the `mediate` function.[6] In the current example, this can be done by setting the `age` covariate to a specific value. To allow the mediation effects to be moderated by age, we set the value of `age` to be 20 in one model and 60 in another model.

[5] Note that `amelidiate` does not support some models and features yet; see `?amelidiate` for details.

[6] If the models include moderator-treatment interactions and yet this option is not specified, then the resulting ACME and direct effect estimates will simply be averages over the empirical distribution of the covariates.

More complicated moderated mediation involving multiple moderators could be specified by expanding the list of the covariates.

```
R> med.age20 <- mediate(med.fit, out.fit, treat = "treat",
+                       mediator = "emo", covariates = list(age = 20), sims = 100)
R> med.age60 <- mediate(med.fit, out.fit, treat = "treat",
+                       mediator = "emo", covariates = list(age = 60), sims = 100)
R> summary(med.age20)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

(Inference Conditional on the Covariate Values Specified in `covariates')

                         Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)             0.0702       0.0101       0.1813    0.04
ACME (treated)             0.0852       0.0144       0.2020    0.04
ADE (control)              0.2275       0.0224       0.4638    0.04
ADE (treated)              0.2425       0.0248       0.4714    0.04
Total Effect               0.3127       0.1122       0.5568    0.00
Prop. Mediated (control)   0.2126       0.0235       0.8238    0.04
Prop. Mediated (treated)   0.2641       0.0334       0.8608    0.04
ACME (average)             0.0777       0.0123       0.1914    0.04
ADE (average)              0.2350       0.0236       0.4676    0.04
Prop. Mediated (average)   0.2383       0.0285       0.8423    0.04

Sample Size Used: 265


Simulations: 100

R> summary(med.age60)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

(Inference Conditional on the Covariate Values Specified in `covariates')

                         Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)            0.07703      0.01058      0.13799    0.04
ACME (treated)            0.06900      0.00919      0.13829    0.04
ADE (control)            -0.08905     -0.22558      0.05295    0.28
ADE (treated)            -0.09708     -0.24478      0.05592    0.28
Total Effect             -0.02005     -0.17471      0.14057    0.78
Prop. Mediated (control) -0.52540     -8.25181     17.47875    0.78
```

```
Prop. Mediated (treated) -0.43131     -7.16792     16.01512    0.78
ACME (average)            0.07302      0.00989      0.13905    0.04
ADE (average)            -0.09306     -0.23236      0.05453    0.28
Prop. Mediated (average) -0.47836     -7.70987     16.74694    0.78

Sample Size Used: 265


Simulations: 100
```

Thus, the researcher receives two different sets of output. In the first output, the average mediation effect is estimated for those who are 20 years old. In contrast, the second output applies to those who are 60 years old.

The second approach to moderated mediation consists of directly testing the statistical significance of the difference in the ACME and ADE between two chosen levels of pre-treatment covariates. This analysis is conducted via the `test.modmed` function. For example, the following syntax can be used to test whether the ACME and ADE significantly differ between the subjects who are 20 years old and those who are 60 years old.

```
R> med.init <- mediate(med.fit, out.fit, treat = "treat", mediator = "emo", sims=2)
R> test.modmed(med.init, covariates.1 = list(age = 20),
+                        covariates.2 = list(age = 60), sims = 100)

        Test of ACME(covariates.1) - ACME(covariates.2) = 0

data:  estimates from med.init
ACME(covariates.1) - ACME(covariates.2) = 0.008, p-value = 0.92
alternative hypothesis: true ACME(covariates.1) - ACME(covariates.2) is not equal to 0
95 percent confidence interval:
 -0.1075738  0.1249199


        Test of ADE(covariates.1) - ADE(covariates.2) = 0

data:  estimates from med.init
ADE(covariates.1) - ADE(covariates.2) = 0.3027, p-value = 0.02
alternative hypothesis: true ADE(covariates.1) - ADE(covariates.2) is not equal to 0
95 percent confidence interval:
 0.04676954 0.59796646
```

Unlike the first approach, the initial `mediate` fit does not need the `covariates` argument set to specific values; the choice of covariate levels is made directly in the call to the `test.modmed` function. Note that the initial `mediate` call does not require a large number of simulation draws, for the actual calculation of uncertainty for the test happens within the `test.modmed` function.

3.3. Non-binary treatment variables

Experimental manipulations are often complex and go beyond simple treatment and control conditions. In the framing experiment, for example, the researchers actually used a 2×2 factorial design. That is, each subject was exposed to two different binary treatments, yielding four different experimental manipulations. In the analysis presented above, we have focused on a comparison of one of these conditions relative to the other three conditions. The `mediate` function, however, has the capability to handle more complex experimental contrasts, which can be represented by a non-binary treatment variable.

Here, instead of using the binary `treat` variable, we use a variable named `cond`, which records which of the four conditions the subject was randomly exposed to. Using the `control.value` and `treat.value` options, the user can calculate the specific contrast of interest. For example, the comparison between the second and third conditions can be done with the following code.

```
R> med.fit <- lm(emo ~ cond + age + educ + gender + income, data = framing)
R> out.fit <- glm(cong_mesg ~ emo + cond + age + educ + gender + income,
+                 data = framing, family = binomial("probit"))
R> med23.out <- mediate(med.fit, out.fit, treat = "cond", mediator = "emo",
+                       control.value = 2, treat.value = 3, sims = 100)
R> summary(med23.out)
```

Similarly, the researcher can compare the first and fourth experimental conditions via the following syntax,

```
R> med14.out <- mediate(med.fit, out.fit, treat = "cond", mediator = "emo",
+                       control.value = 1, treat.value = 4, sims = 100)
R> summary(med14.out)
```

Nothing changes in the format of the output, but the contrasts differ depending on the categories chosen for comparison by the researcher. In the case of a continuous treatment variable, the researcher would specify two values of the treatment to make the contrast (Imai et al. 2010a). For example, the causal mediation effects can be defined for any two levels of the treatment,

$$\delta_i(t; t_1, t_0) \quad \equiv \quad Y_i(t, M_i(t_1)) - Y_i(t, M_i(t_0)), \tag{7}$$

where $t_1 \neq t_0$. The corresponding average causal mediation effect is defined as $\bar{\delta}(t; t_1, t_0) \equiv \mathbb{E}(\delta_i(t; t_1, t_0))$. Thus, the researcher can set `control.value` to t_0 and `treat.value` to t_1. The researcher may also vary the value of t_1, while fixing the base line value of t_0, to examine how the ACME changes as the function of t_1.

3.4. Sensitivity analysis for sequential ignorability

Sequential ignorability is a strong assumption, and therefore a sensitivity analysis is recommended. The **mediation** package allows the researcher to conduct a sensitivity analysis for the possible existence of unobserved pre-treatment covariates. Specifically, the output of the `mediate` function can be passed to the `medsens` function, which then computes the values of causal quantities as a function of sensitivity parameters. Both `summary` and `plot` functions are available for sensitivity analysis, and they display the results in a tabular and graphical form, respectively. Since derivation of sensitivity formulas must be done on a case-by-case

	Outcome model types	
Mediator model types	Linear	Binary probit
Linear	✓	✓
Binary probit	✓	–

Table 2: The types of models that can be handled by `medsens` for sensitivity analysis.

basis, the range of options for conducting sensitivity analyses is somewhat limited. Table 2 gives the model combinations currently supported by the `medsens` function.

In our running example, after computing the ACME, we conduct a sensitivity analysis by passing the object from `mediate` to the `medsens` function. We first choose as the sensitivity parameter the correlation ρ between the residuals of the mediator and outcome regressions (Imai et al. 2010a,c). If there exist unobserved pre-treatment confounders which affect both the mediator and the outcome, we expect that the sequential ignorability assumption is violated and ρ is no longer zero. The sensitivity analysis is conducted by varying the value of ρ and examining how the estimated ACME changes. The following syntax can be used to conduct this analysis,

```
R> med.fit <- lm(emo ~ treat + age + educ + gender + income, data = framing)
R> out.fit <- glm(cong_mesg ~ emo + treat + age + educ + gender + income,
+                 data = framing, family = binomial("probit"))
R> med.out <- mediate(med.fit, out.fit, treat = "treat", mediator = "emo",
+                     robustSE = TRUE, sims = 100)
R> sens.out <- medsens(med.out, rho.by = 0.1, effect.type = "indirect", sims = 100)
R> summary(sens.out)


Mediation Sensitivity Analysis: Average Mediation Effect

Sensitivity Region: ACME for Control Group

     Rho ACME(control) 95% CI Lower 95% CI Upper R^2_M*R^2_Y* R^2_M~R^2_Y~
[1,] 0.3        0.0058      -0.0055       0.0206         0.09       0.0493
[2,] 0.4       -0.0095      -0.0285       0.0024         0.16       0.0877

Rho at which ACME for Control Group = 0: 0.3
R^2_M*R^2_Y* at which ACME for Control Group = 0: 0.09
R^2_M~R^2_Y~ at which ACME for Control Group = 0: 0.0493


Sensitivity Region: ACME for Treatment Group

     Rho ACME(treated) 95% CI Lower 95% CI Upper R^2_M*R^2_Y* R^2_M~R^2_Y~
[1,] 0.3        0.0066      -0.0069       0.0222         0.09       0.0493
[2,] 0.4       -0.0118      -0.0351       0.0026         0.16       0.0877

Rho at which ACME for Treatment Group = 0: 0.3
```

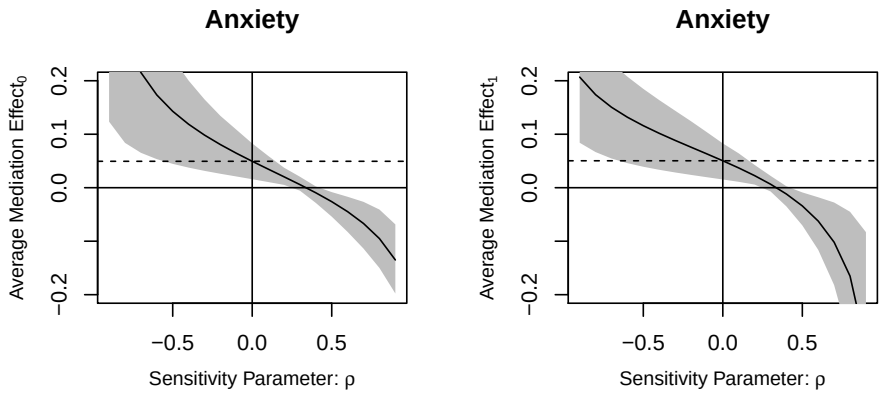

Figure 3: Graphical display of results from the `medsens` function. Results as a function of ρ.

```
R^2_M*R^2_Y* at which ACME for Treatment Group = 0: 0.09
R^2_M~R^2_Y~ at which ACME for Treatment Group = 0: 0.0493
```

where `rho.by = 0.1` specifies that ρ will vary from -0.9 to 0.9 by 0.1 increments, and `effect.type = "indirect"` means that sensitivity analysis is conducted for the ACME. Alternatively, specifying `effect.type = "direct"` performs sensitivity analysis for the ADE and `"both"` returns sensitivity analysis for the ACME and ADE.

The tabular output from the `summary` function displays the values of ρ at which the confidence intervals contain zero for the ACME. For both the control and treatment conditions, the confidence intervals for the ACME contain zero when ρ equals 0.3 and 0.4. An alternative but mathematically equivalent way to conduct sensitivity is in terms of the product of R^2 (or coefficients of determination) statistics from the mediator and outcome models. Discussed in more detail elsewhere (Imai et al. 2010c, 2011, 2010a), the first row captures the point at which the ACME is 0 as a function of the proportions of residual variance in the mediator and outcome explained by the hypothesized unobserved confounder. The second line uses the total variance instead of residual variance. We use R^{*2} for residual variance and $\tilde{R}^2$ for total variance. For example, when the product of the original variance explained by the omitted confounding is .049 the point estimate for ACME would be 0.

A graphical display is often more intuitive and useful for the sensitivity analysis, especially for the R^2 interpretations. This can be done, as before, by passing the object from the `medsens` function to the `plot` function. The `plot` function allows the researcher to graphically summarize the results of sensitivity analysis either in terms of ρ (`sens.par = "rho"`) or R^2 statistics (`sens.par = "R2"`).

```
R> plot(sens.out, sens.par = "rho", main = "Anxiety", ylim = c(-0.2, 0.2))
```

When using the R^2 statistic version of sensitivity analysis the user must specify whether the hypothesized confounder affects the mediator and outcome variables in the same direction or in different directions. This matters because the sensitivity analysis is in terms of the product of R^2 statistics. In the current example, we assume that the confounder influences both

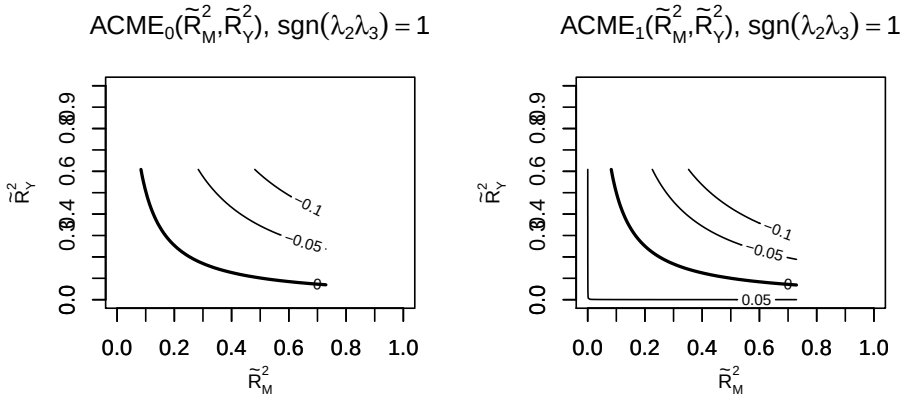

Figure 4: Graphical display of results from the `medsens` function. Results as a function of $\tilde{R}^2$.

variables in the same direction by setting `sign.prod = "positive"` (rather than `sign.prod = "negative"`). Here, we plot the total variance version of the sensitivity analysis. The bold line represents the various combinations of the R^2 statistics where the ACME would be 0 (in this case the product equals .049). The graphical display also presents the corresponding contour plots for other products of the R^2 statistics.

```
R> plot(sens.out, sens.par = "R2", r.type = "total", sign.prod = "positive")
```

4. Causal mediation analysis of multilevel data

As of version 4.2, the **mediation** package supports causal mediation analysis of multilevel data via the `lmer` and `glmer` functions in the **lme4** package (Bates, Maechler, Bolker, and Walker 2014). Researchers are often interested in analyzing data where individual observations such as students, patients, and employees are clustered within groups such as schools, hospitals, and companies. Data on individuals may be correlated within groups, but also different groups may have different data generating processes. Multilevel models take into account such heterogeneity within and between groups simultaneously.

Mediation analysis of multilevel data can be categorized into various types depending on whether the treatment, mediator and outcome variables are each measured at the individual or group level (see Krull and MacKinnon 2001; Zhang, Zyphur, and Preacher 2009). Regardless of these types, researchers can use the `mediate` function to analyze multilevel data by choosing appropriate statistical models for the mediator and outcome variables. In this section, we illustrate the use of our package for multilevel data by focusing on two types of data structure: (1) the treatment is assigned at the group level whereas the mediator and outcome are measured at the individual level, and (2) both the treatment and mediator are group-level variables while the outcome is recorded at the individual level. Other combinations of data levels can be handled via straightforward modifications to the syntax used in these examples.[7]

[7]We note that as of the writing of this article the lme4 package is known to generate slightly different

To illustrate the usage, we analyze data from the Education Longitudinal Study (2002)[8] where students are clustered within schools. The **mediation** package contains two related data sets. The `student` data set contains both student- and school-level variables organized at the student level. The `school` data set only contains school-level variables, such that the number of observations in this data set equals the number of unique levels of the school identifier variable (`SCH_ID`) in the `student` data set. As explained below in detail, the group-level data set (`school`) is required only when we analyze the data where both the treatment and the mediator are group-level variables.

4.1. Group-level treatment and individual-level mediator

First, consider the case where the treatment is a group-level variable but the mediator and outcome variables are measured at the individual level. In this case, we only need the student-level data set,

```
R> data("student", package = "mediation")
```

Here, we analyze as an example whether a school is Catholic or not (`catholic`) affects a student's likelihood of fighting (`fight`) at the school, and hypothesize that a student's emotional attachment to the school (`attachment`) functions as the causal mechanism. That is, we postulate that students in a Catholic school may have an increased sense of attachment to their school, which may in turn decrease their likelihood of getting involved in a fight. We model these causal processes using the following hierarchical logistic-normal regression model for the (binary) mediator,

$$\begin{aligned} \mathsf{P}(M_{ij} = 1) &= \text{logit}^{-1}\left(\alpha_j + \gamma^\top X_{ij}\right), \\ \alpha_j &= \alpha + \beta T_j + \varepsilon_j, \end{aligned}$$

where i and j are student and school indicators, respectively, ε_j is a normally distributed group-level stochastic error with mean zero, and X_{ij} represents the vector of student-level pre-treatment covariates (`gender`, `income` and `pared`). Likewise, we use the following model for the (binary) outcome,

$$\begin{aligned} \mathsf{P}(Y_{ij} = 1) &= \text{logit}^{-1}\left(\lambda_j + \phi_j M_{ij} + \zeta^\top X_{ij}\right), \\ \lambda_j &= \lambda + \psi T_j + \upsilon_j, \\ \phi_j &= \phi + \theta T_j + \nu_j, \end{aligned}$$

where υ_j and ν_j are group-level errors jointly bivariate normally distributed with mean zero. If desired, more complex data generating processes can be assumed (with appropriate changes

random number draws across computing platforms (Windows, Mac, etc.) for a given seed which given the simulation method used can generate small numerical differences in some cases.

[8]To protect the anonymity of the individuals involved in the study, we generated hypothetical individual-level variables via multiple imputation. The results reported below do not take into account any statistical uncertainty due to the imputation procedure and should thus be regarded only as illustration. The original data can be obtained from *Education Longitudinal Study (ELS), 2002: Base Year (ICPSR 4275)* by the United States Department of Education, National Center of Education Statistics. http://www.icpsr.umich.edu/icpsrweb/ICPSR/studies/4275.

in the syntax for the models below), such as allowing for group-varying slopes on the treatment variable or incorporating group-level pre-treatment covariates.

Now, note that these two models can be equivalently written as follows,

$$\mathsf{P}(M_{ij} = 1) \;=\; \text{logit}^{-1}\left((\alpha + \varepsilon_j) + \beta T_j + \gamma^\top X_{ij}\right),$$

and

$$\mathsf{P}(Y_{ij} = 1) \;=\; \text{logit}^{-1}\left((\lambda + \upsilon_j) + \psi T_j + (\phi + \nu_j) M_{ij} + \theta T_j M_{ij} + \zeta^\top X_{ij}\right),$$

which can both be estimated via the `glmer` function,

```
R> library(lme4)
R> med.fit <- glmer(attachment ~ catholic + gender + income + pared + (1|SCH_ID),
+                   family = binomial(link = "logit"), data = student)
R> out.fit <- glmer(fight ~ catholic*attachment +
+                           gender + income + pared + (1 + attachment|SCH_ID),
+                   family = binomial(link = "logit"), data = student)
```

These fitted objects can then be fed into the `mediate` function in the usual manner.

```
R> med.out <- mediate(med.fit, out.fit, treat = "catholic", mediator = "attachment",
+                     sims = 100)
R> summary(med.out)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

Mediator Groups: SCH_ID

Outcome Groups: SCH_ID

Output Based on Overall Averages Across Groups

                         Estimate 95% CI Lower 95% CI Upper p-value
ACME (control)           -0.00393     -0.00685     -0.00135       0
ACME (treated)           -0.00392     -0.00777     -0.00142       0
ADE (control)            -0.02564     -0.04068     -0.00605       0
ADE (treated)            -0.02563     -0.03975     -0.00639       0
Total Effect             -0.02956     -0.04432     -0.01182       0
Prop. Mediated (control)  0.12983      0.05464      0.31910       0
Prop. Mediated (treated)  0.12167      0.05060      0.36426       0
ACME (average)           -0.00392     -0.00693     -0.00156       0
ADE (average)            -0.02564     -0.04021     -0.00622       0
Prop. Mediated (average)  0.12575      0.05705      0.33895       0

Sample Size Used: 9679
```

```
Simulations: 100
```

The estimated mediation, direct, and total effects are all significantly different from zero. The results suggest that the school-level treatment (`catholic`) increases the value of the individual-level mediator (`attachment`), which in turn decreases the value of the outcome (`fight`), and also that the treatment decreases the value of the outcome directly or in different causal paths.

4.2. Group-level treatment and mediator

Next, consider the case where both the treatment and mediator are group-level variables while the outcome is measured at the individual level. In this case, we need the second data set containing only the group-level variables,

```
R> data("school", package = "mediation")
```

Note that the group-level data set must also contain the group indicator used in the individual-level data set under the same variable name (`SCH_ID` in our running example). The current version of `mediate` also requires that the model frames of the mediator and outcome models contain the exact same set of groups, which becomes important when each model contains different covariates and some groups drop out of the model frames due to missingness.

As an illustration, we investigate the effects of school-level economic condition (`free`; proportion of students who receive free lunch) on students' tardiness (`late`; days per semester they are late for school). As a causal path, we postulate that school-level poverty negatively impacts school-level morale (`smorale`), which in turn increases tardiness among students. We use the following hierarchical regressions to model the hypothesized causal mechanism,

$$\begin{aligned} M_j &= \alpha + \beta T_j + \gamma^\top V_j + \varepsilon_j, \\ Y_{ij} &= \lambda_j + \zeta^\top X_{ij} + \upsilon_{ij}, \\ \lambda_j &= \lambda + \theta T_j + \phi M_j + \kappa^\top V_j + \nu_j, \end{aligned}$$

where V_j is the vector of school-level covariates (`catholic` and `coed`), X_{ij} is the vector of student-level covariates (`gender`, `income` and `pared`), and ε_j, υ_{ij} and ν_j are each normally distributed stochastic errors with mean zero. Again, more complex models can be used (e.g., adding a treatment-mediator interaction term to the outcome model) if desired.

In this case, the mediator model is solely composed of the school-level variables and fixed coefficients. Hence the mediator model can be fit via the `lm` function using the school-level data set,

```
R> med.fit <- lm(smorale ~  free + catholic + coed, data = school)
```

and the outcome model, which can be equivalently written as,

$$Y_{ij} = (\lambda + \upsilon_j) + \theta T_j + \phi M_j + (\gamma^\top + \kappa^\top) V_j + \zeta^\top X_{ij} + \upsilon_{ij},$$

can be estimated with the `lmer` function and the student-level data set,

```
R> out.fit <- lmer(late ~ free + smorale + catholic + coed +
+                         gender + income + pared + (1|SCH_ID),
+                 data = student)
```

These fitted model objects can then be passed to the `mediate` function. Since the treatment variable is a continuous variable, we use the values of 3 and 4 as the control and treatment values, respectively, and estimate the quantities of interest in terms of these values.

```
R> med.out <- mediate(med.fit, out.fit, treat = "free", mediator = "smorale",
+                     control.value = 3, treat.value = 4, sims = 100)
R> summary(med.out)

Causal Mediation Analysis

Quasi-Bayesian Confidence Intervals

Mediator Groups:

Outcome Groups: SCH_ID

Output Based on Overall Averages Across Groups

                Estimate 95% CI Lower 95% CI Upper p-value
ACME            0.007094     0.002554     0.012211    0.00
ADE             0.020356    -0.000729     0.038802    0.06
Total Effect    0.027450     0.005651     0.047817    0.02
Prop. Mediated  0.260223     0.080815     0.851923    0.02

Sample Size Used: 9679


Simulations: 100
```

The estimated mediation effect is significantly different from zero, suggesting that the school-level treatment (`free`) decreases the value of the school-level mediator (`smorale`), which in turn increases the value of the outcome (`late`).

We conclude this section by providing more details about the current version of our package for multilevel mediation analysis. First, the `summary` function can produce estimates of group-specific effects by adding the `output` argument, which can be set to either `"bygroup"` or `"byeffect"`. In the above example, `summary(med.out, output = "bygroup")` produces the output organized by schools, and `summary(med.out, output = "byeffect")` produces the output organized into quantities of interest. Group-specific effects can also be graphically displayed by `plot(med.out, group.plots = TRUE)`. Second, the `mediate` function allows researchers to specify different groups in the mediator and outcome models (nested or non-nested). For example, it may be reasonable to assume that the mediator variable is correlated within schools but the outcome variable is clustered at the district level. In such a case, the

`group.out` argument for the `mediate` function allows researchers to choose the group into which the estimated group-specific effects are aggregated.

The current version of the package also has some limitations for multilevel mediation analysis. First, it only allows for one group type for each model. For example, it is not possible to let coefficients of the mediator (or outcome) model vary not only for schools but also for districts. Second, the bootstrap-based uncertainty estimates for the mediation effects are not yet available. Third, the `medsens` function for sensitivity analysis cannot be applied to the `mediate` outputs based on multilevel regression models. Future updates may add these missing functionalities. Finally, it is important to reiterate that the validity of the estimates crucially rests on Assumption 1, regardless of whether hierarchical models are fitted to the data or not.

5. Design-based causal mediation analysis

An alternative approach to model-based inference is to use different research designs that are specifically designed for identifying causal mechanisms. Imai et al. (2013) propose several such designs and describe the assumptions required for the identification of causal mediation effects under each of the designs. In this section we briefly illustrate how to use our software to calculate the estimates of the quantities of interest under each design.

5.1. Single experiment design

The single experiment design randomizes the treatment variable and measures the mediating and outcome variables. In Section 3, we discussed estimation functions that can be used with parametric and semi-parametric models. If the researchers wish to pursue a completely non-parametric approach the **mediation** package offers two options via the `mediate.sed` function. First, the researchers can continue to make the sequential ignorability assumption and non-parametrically estimate the ACME. This approach works only when the mediator variable is discrete. Second, the sharp bounds on the ACME can be computed under the assumption that only the treatment is randomized. Imai et al. (2013) derive the bounds in the case with all binary variables (treatment, mediator, and outcome) and show that, unfortunately, the bounds are never informative about the sign of the ACME (i.e., they always include 0).

Most mediation analysis proceeds under the sequential ignorability assumption. Those analyses also tend to be model-based, but they need not be. Imai et al. (2010c) outline a design-based estimator for the ACME for when the mediator is discrete. This estimator for the ACME is fully nonparametric. One drawback to this estimator is that one can encounter mediator-treatment combinations for which there are no subjects because of data sparsity. Standard error calculation for this estimator is based on either the Delta method or the nonparametric bootstrap.

The `mediate.sed` function requires the names of the outcome, mediator, and treatment variables, along with the name of the data frame that contains these variables. When `SI = TRUE`, the function will return the point estimates under the sequential ignorability assumption, and otherwise the results will be a set of sharp bounds for the ACME. The method for inference also differs slightly from the `mediate` function. When `boot = TRUE` the bootstrap is used, but when `boot = FALSE`, the Delta method is used to compute standard errors.

Below, we present an example using the framing data from Brader et al. (2008). The treatment

variable is the same as before, i.e., `treat`, and the mediator is `anx`, which refers to a subject's reported level of anxiety. This four level measure is one component of the `emo` variable that was previously used as the mediator and in the data all treatment-mediator combinations are present (a requirement for the estimator). The outcome variable in this example is `english` and measures on a four point scale how much someone supports English only laws, from strongly support to strongly oppose. Note that the `mediate.sed` function only takes numeric variables as arguments. Variables that are stored as factors must be converted to numeric variables as we show below.

```
R> framing$english <- as.numeric(framing$english)
R> framing$anx <- as.numeric(framing$anx)
R> sed.est <- mediate.sed("english", "anx", "treat", data = framing, SI = TRUE,
+                         boot = TRUE, sims = 100)
R> summary(sed.est)

Design-Based Causal Mediation Analysis

Single Experiment Design with Sequential Ignorability

Confidence Intervals Based on Nonparametric Bootstrap

               Estimate 95% CI Lower 95% CI Upper
ACME (control)  0.10212     -0.56766        1.011
ACME (treated)  0.07066     -0.21566        0.379

Sample Size Used:  265
```

The results from the `summary` function display the mediation effects along with the default 95% confidence intervals. In this example both $\bar{\delta}(0)$ and $\bar{\delta}(1)$ are not significantly different from 0.

5.2. Parallel design

An alternative to the single experiment design is the "parallel design" proposed by Imai et al. (2013). In this design there are two separate experiments that are run in parallel with subjects randomly assigned to one of the two experiments. The first experiment follows the single experiment design. In the second experiment, subjects are randomly assigned to treatment or control. Then, a randomly selected set of subjects in each condition is assigned a value of the mediating variable. A key assumption of this design is that the manipulation of the mediating variable is possible and has no direct effect on the outcome variable.

Under the parallel design, the ACME is not point identified without an additional assumption. The **mediation** package offers two options via the `mediate.pd` function. First, the researchers can assume no interaction between the treatment and mediating variables by setting `NINT = TRUE`. In this case, the `mediate.pd` function will calculate the ACME along with its bootstrap confidence intervals. Second, the assumption of no-interaction between treatment and mediator can be dropped via `NINT = FALSE`, and then the sharp bounds can be calculated for the ACME. These bounds may be informative about the sign (i.e., do not cover 0) and are

always narrower compared to the bounds under the single experiment design where the only assumption is randomization of the treatment.

For illustration, we simulated data based on the media framing experiment by Brader et al. (2008) by creating a population distribution of potential mediators and outcomes (see Imai et al. (2013) for more details). We then sampled 1000 cases from this distribution. In this example, `out` represents the outcome variable (immigration attitudes), `med` represents the mediator (anxiety), and `ttt` represents the treatment. All variables are binary. The variable `manip` represents whether the subject had the mediator manipulated (-1 if mediator is manipulated down, 0 if no manipulation, and 1 if manipulated up). First, the no-interaction assumption is made and options for the number of bootstrap simulations and confidence intervals are specified. In this case, the mediation effect is estimated at -0.12 with 95% confidence intervals spanning $[-0.21, -0.03]$. In the second example, the no interaction assumption is dropped and the sharp bounds are calculated to span $[-0.3, 0.3]$ for the control condition and $[0.2, 0.77]$ for the treatment condition.

```
R> data("boundsdata", package = "mediation")
R> pd <- mediate.pd("out", "med", "ttt", "manip", boundsdata,
+                   NINT = TRUE, sims = 100, conf.level = 0.95)
R> summary(pd)

Design-Based Causal Mediation Analysis

Parallel Design (with No Interaction Assumption)

     Estimate 95% CI Lower 95% CI Upper
ACME  -0.1236      -0.2198       -0.035

Sample Size Used:  1000

R> pd1 <- mediate.pd("out", "med", "ttt", "manip", boundsdata,
+                    NINT = FALSE)
R> summary(pd1)

Design-Based Causal Mediation Analysis

Parallel Design (Interaction Allowed)

               Lower Bound Upper Bound
ACME (control)     -0.3207       0.330
ACME (treated)      0.2006       0.768

Sample Size Used:  1000
```

5.3. Parallel encouragement design

In many situations, perfect manipulation of the mediating variable may be difficult. In the parallel encouragement design, subjects are split into two separate experiments. The first

experiment is based on the single experiment design. In the second experiment subjects are randomly assigned to the treatment and control conditions and then, within each condition, a subset of subjects are randomly encouraged to have a high or low value of the mediator. Both the mediator and outcome variable are then measured. The `mediate.ped` function reports the sharp bounds on two estimands. First is the ACME and second is the ACME for the "compliers" who respond to the encouragement. The calculation of these bounds is accomplished via a standard linear programming approach as discussed in Imai et al. (2013).

The parallel encouragement design requires the analyst to specifically design some form of encouragement. The functionality of the `mediate.ped` closely mirrors that of `mediate.sed`. The key difference is that the analyst must also include an indicator for encouragement. For illustration, we simulated data based on the media framing experiment by Brader et al. (2008). We did this by creating a population distribution of potential mediators and outcomes, and compliance types. We then randomly draw the joint probabilities of the causal types and assign an encouragement status for those in the encouragement condition (see Imai et al. (2013) for more details). Based on the encouragement condition and encouragement status (`enc`, -1 if mediator is encouraged down, 0 if no encouragement, and 1 if encouraged up), the observed binary values of the mediator (`med.enc`) and outcome (`out.enc`) are determined. Using this simulated data we can then pass it to the `mediate.ped` function for the parallel encouragement design.

```
R> data("boundsdata", package = "mediation")
R> ped <- mediate.ped("out.enc", "med.enc", "ttt", "enc", boundsdata)
R> summary(ped)

Design-Based Causal Mediation Analysis

Parallel Encouragement Design

                          Lower Bound Upper Bound
Population ACME (control)    -0.43407       0.324
Complier ACME (control)      -0.14649       0.208
Population ACME (treated)    -0.02014       0.743
Complier ACME (treated)       0.01137       0.707

Sample Size Used:  1000
```

Here, the results from `mediate.ped` function are a set of sharp bounds. We see that for the compliers, the sharp bounds on ACME under the treatment condition are informative as they do not cross 0.

5.4. Crossover encouragement design

The fourth experimental design included in the **mediation** package is the crossover encouragement design. Under this design, subjects are exposed to two experiments, with each subject participating in each experiment. In the first experiment, the treatment variable is randomized and the mediator and outcome variables observed. In the second experiment, the treatment condition is set to the opposite value from the first period, but an encouragement

is given to a randomly selected set of subjects so that the mediator variable will take on the value observed in the first experiment. Under this design, the ACME is point identified for the set of subjects that are able to have their mediator value manipulated (known as "pliable units"). A crucial identification assumption is that the first experiment does not influence behavior in the second experiment. For this experimental design the `mediate.ced` function calculates point estimates and the bootstrap is used for estimates of uncertainty.

For illustration, we simulated data based on the identification assumptions necessary for this design. `Y2` is the value of the outcome variable in the second experiment, `M1` and `M2` are the mediator values for the first and second experiment, `T1` is the value of the treatment in the first experiment, and `Z` indicates whether the subject's mediator value in the second experiment is encouraged to take on the value opposite to that observed in the first experiment. All variables are binary.

```
R> data("CEDdata", package = "mediation")
R> ced <- mediate.ced("Y2", "M1", "M2", "T1", "Z", CEDdata, sims = 100)
R> summary(ced)

Design-Based Causal Mediation Analysis

Crossover Encouragement Design

                       Estimate 95% CI Lower 95% CI Upper
Pliable ACME (control)  0.09069     -0.11769        0.300
Pliable ACME (treated)  0.11935     -0.05790        0.313

Sample Size Used:  2000
```

The results from the `mediate.ced` function are point estimates and confidence intervals for the ACME under the treatment and control conditions. These estimates apply only to the pliable units. In this example, both values of the ACME are positive but the 95% confidence intervals overlap with zero.

6. Analysis of causally dependent multiple mechanisms

Our discussion so far has focused on a single mediator, M. Frequently, however, researchers take measurements for more than one mediating variable. Accounting for alternative mechanisms is indeed crucial for the identification of the mechanism of primary interest, especially when such mechanisms are causally not independent. This is because the alternative dependent mediators affect both the mediator of primary interest and the outcome variable, which, by definition, violates the sequential ignorability assumption (Assumption 1).

6.1. The methodology

Imai and Yamamoto (2013) develop methods for dealing with multiple mediators based on the current framework. We briefly review this framework. First, in the case of causally unrelated multiple mediators, it turns out that there is no need to fundamentally modify the current

framework or estimation procedure. To see this, suppose that there are multiple causally unrelated mediators, and one is interested in estimating the causal mediation effects with respect to each of them. In this scenario, note that for each mediator, the other mediators are neither pre-treatment nor post-treatment confounders (since by construction they have no causal effect on the mediator of interest). Therefore, one can consistently estimate the desired effects by simply applying the `mediate` function successively for the mediators as explained in Section 3, ignoring the existence of the other, causally unrelated, mediators each time. Likewise, sensitivity analysis via the `medsens` function can be conducted for the mediators of interest in the usual fashion. The `mediations` function can be useful for such analysis.

Second, when the multiple mediators are causally related (or equivalently, when one mediator acts as a post-treatment confounder for the other mediator on the outcome), we need to expand the notational framework, and the analysis requires new assumptions. Let $W_i(t)$ denote the vector of the potential values of those alternative mediators given treatment status t. To allow the causal dependence of both the primary mediator and outcome on W, we write the potential mediator and outcome as $M_i(t, w)$ and $Y_i(t, m, w)$, respectively. The observed values of these potential response variables can then be expressed as $W_i = W_i(T_i)$, $M_i = M_i(T_i, W_i(T_i))$, and $Y_i = Y_i(T_i, M_i(T_i, W_i(T_i)), W_i(T_i))$. The causal mediation effects can now be re-expressed using this notation as,

$$\delta_i(t) \quad = \quad Y_i(t, M_i(1, W_i(1)), W_i(t)) - Y_i(t, M_i(0, W_i(0)), W_i(t)),$$

for $t = 0, 1$. Note that this quantity represents the treatment effects that are transmitted through the mediator of primary interest M_i, irrespective of whether they also come through the alternative mediators W_i or not. Therefore, the quantity of interest remains unchanged from the previous sections, except that the existence of the other mediators are now explicitly taken into consideration.

The framework of Imai and Yamamoto (2013) is based on the following varying coefficient linear structural equations model,

$$M_i(t, w) \quad = \quad \alpha_2 + \beta_{2i} t + \xi_{2i}^\top w + \mu_{2i}^\top tw + \lambda_{2i}^\top x + \varepsilon_{2i}, \tag{8}$$

$$Y_i(t, m, w) \quad = \quad \alpha_3 + \beta_{3i} t + \gamma_i m + \kappa_i tm + \xi_{3i}^\top w + \mu_{3i}^\top tw + \lambda_{3i}^\top x_i + \varepsilon_{3i}, \tag{9}$$

where $\mathbb{E}(\varepsilon_{2i}) = \mathbb{E}(\varepsilon_{3i}) = 0$ without loss of generality. Although these equations may resemble a traditional linear structural equations model at a first glance, they are considerably more flexible because the coefficients are all allowed to vary arbitrarily across individual units.

Imai and Yamamoto (2013) propose two strategies for the analysis of the average causal mediation effects, $\bar{\delta}(t) \equiv \mathbb{E}(\delta_i(t))$. First, it can be shown that the ACME is point identified under the above model and sequential ignorability (a weaker version allowing for post-treatment confounding; see Imai and Yamamoto) if the *homogeneous interaction* assumption is satisfied. This additional assumption is formally written as,

$$Y_i(1, m, W_i(1)) - Y_i(0, m, W_i(0)) \quad = \quad B_i + Cm$$

for any m. The assumption states that the degree of interaction between the treatment and the primary mediator is constant across individual units, which may or may not be plausible depending on the empirical context.

Second, when this assumption is violated, one can express the sharp bounds on the ACME as functions of a parameter representing the degree of the violation, and conduct a sensitivity

analysis. The sensitivity parameter here is the standard deviation of the coefficient on the treatment-mediator interaction term, i.e.,

$$\sigma \equiv \sqrt{\mathsf{VAR}(\kappa_i)},$$

and the expression for the sharp bounds are given in Imai and Yamamoto (2013, Footnote 6). Researchers can then analyze robustness to the potential violation of the homogeneous interaction assumption by examining how the location and width of the bounds vary as σ changes.

The sensitivity analysis can also be formulated in terms of two alternative sensitivity parameters, both based on coefficients of determination as in the single mediator case (see Section 3.4). Specifically, we use the proportion of the residual or total variance of the outcome variable that would be explained by allowing the heterogeneity in the treatment-mediator interaction in the outcome model. These parameters are formally defined as

$$R^{2*} = \frac{\mathsf{VAR}(\tilde{\kappa}_i T_i M_i)}{\mathsf{VAR}(\eta_{3i}(T_i, M_i, W_i))} \quad \text{and} \quad \widetilde{R}^2 = \frac{\mathsf{VAR}(\tilde{\kappa}_i T_i M_i)}{\mathsf{VAR}(Y_i)}, \tag{10}$$

where $\tilde{\kappa}_i = \kappa_i - \mathbb{E}(\kappa_i)$. Researchers may find these parameters to be easier to interpret in substantive terms, as they represent how important it would be to incorporate the interaction heterogeneity in order to explain the variation in the outcome model. Imai and Yamamoto (2013) show that these parameters have a one-to-one relationship with σ, implying that the ACME can also be written as a function of R^{2*} or $\widetilde{R}^2$.

6.2. Single experiment design

The above framework has been implemented in the **mediation** package as the `multimed` function. The function takes a data frame containing the necessary variables (outcome, primary mediator, alternative mediator, treatment, and pre-treatment covariates if any) and outputs an object of class '`multimed`', a list consisting of estimated bounds along with uncertainty estimates. In the current version, only a single post-treatment confounder is allowed, although the theoretical framework accommodates more than one such confounder.

The functionality of `multimed` differs in important ways from `mediate`. First, there is not a separate function for sensitivity analysis. Instead, a sensitivity analysis is conducted within the function along with the estimates of the mediation effects. Second, the arguments for the `multimed` function are rather different. Here, the names of the outcome (`outcome`), first mediator (`med.main`), second mediator (`med.alt`) and treatment (`treat`) variables are passed to the function along with a vector of the names of the pre-treatment covariates to condition on (`covariates`). In the `multimed` function, inference can only be done with the nonparametric bootstrap.

To illustrate the use of the function we revisit the media framing example in Section 3. Here, we use a different outcome variable `immigr`, which is a five category measure of whether immigration should be increased or decreased (treated as a continuous measure for the purpose of illustration). The main mediator is the same composite measure of anxiety, `emo`, and the treatment and pre-treatment covariates are defined as before. We now introduce an alternative mediator `p_harm`, which is an eight category measure of the perceived economic harm of immigrants. The reasoning behind the inclusion of this variable is that the media framing treatment may also affect participants' opinion about immigrants by changing their factual

belief about the economic impact of increased immigration, which may also affect the level of anxiety and therefore confound the mediator-outcome relationship.

```
R> Xnames <- c("age", "educ", "gender", "income")
R> m.med <- multimed(outcome = "immigr", med.main = "emo", med.alt = "p_harm",
+                    treat = "treat", covariates = Xnames,
+                    data = framing, sims = 100)
R> summary(m.med)

Causal Mediation Analysis with Confounding by an Alternative Mechanism

Estimates under the Homogeneous Interaction Assumption:
               Estimate 95% CI Lower 95% CI Upper
ACME (treated)  0.06447     -0.09734         0.23
ACME (control)  0.12397      0.01555         0.23
ACME (average)  0.10870      0.00618         0.21
Total Effect    0.41752      0.16818         0.62

Sensitivity Analysis:
Values of the sensitivity parameters at which ACME first crosses zero:
               sigma(bounds) sigma(CI) R2s(bounds) R2s(CI) R2t(bounds) R2t(CI)
ACME (treated)        0.0299    0.0000      0.0300  0.0000      0.0178    0.00
ACME (control)        0.0489    0.0173      0.0800  0.0100      0.0474    0.01
ACME (average)        0.0423    0.0173      0.0600  0.0100      0.0356    0.01
```

The `summary` function produces two tables. The first table shows the estimated ACME and total treatment effect and their confidence intervals (default at 95%) under the homogeneous interaction assumption. Three variants of the ACME are shown: the ACME conditional on the treatment group, the control group, and the weighted average of the two with the weights being equal to the proportions of the treatment and control groups. The second table shows key summary results from the sensitivity analysis with respect to possible heterogeneity in treatment-mediator interactions. Specifically, the table presents the values of σ (column 1), R^{2*} (column 3), and $\widetilde{R}^2$ (column 5) at which the estimated ACMEs equal zero. The remaining columns (2, 4 and 6) show those values for the confidence bands of the three ACMEs.

The results from the `multimed` function can also be analyzed graphically using the `plot` function. One can produce two types of plots, corresponding to the two tables in the `summary` output. First, one can plot the point estimates under the homogeneous interaction assumption by setting the `type` argument to `"point"`, as shown below. The output is in Figure 5.

```
R> plot(m.med, type = "point")
```

Second, the results from the sensitivity analysis with respect to σ, R^{2*} or $\widetilde{R}^2$ can be plotted. In this case, the `type` argument can be used to specify which parameter(s) the estimated ACME should be plotted against. The possible values are `"sigma"`, `"R2-residual"`, or `"R2-total"`. One can also choose the types of the ACME from `"treated"`, `"control"` and `"average"` via the `tgroup` argument. In the example below, we plot the estimated ACME for both treatment and control conditions as a function of σ and $\widetilde{R}^2$. The output is in Figure 6.

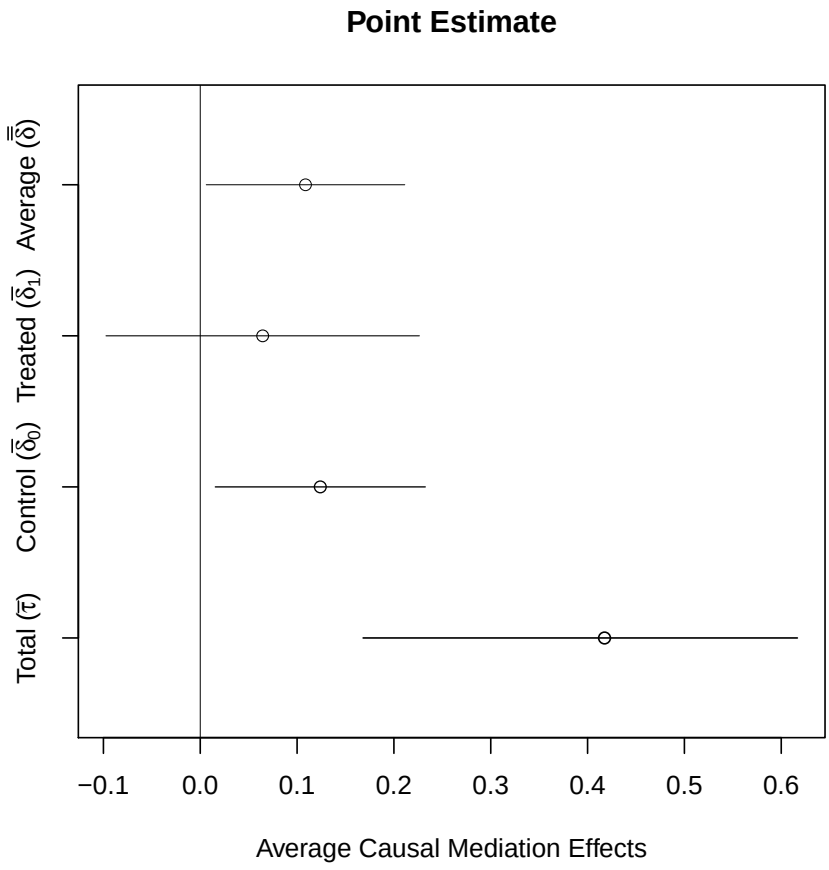

Figure 5: Graphical summary of the results from the `multimed` function under the homogeneous interaction assumption.

```
R> plot(m.med, type = c("sigma", "R2-total"), tgroup = c("treated", "control"))
```

6.3. Parallel design

Imai and Yamamoto (2013, Section 7) show that the above framework can also be applied to the data collected under the parallel design. As discussed in Section 5.2, the parallel design consists of two separate experiments to which subjects are randomly selected. In one experiment, only the treatment is randomized and the researcher observes the mediator and outcome variables, whereas in the other experiment both the treatment and mediator are randomly manipulated and the outcome variable is measured and recorded.

Unlike the single experiment design, one need not assume any kind of sequential ignorability under the parallel design. This is due to the existence of the second experimental group where both the treatment and mediator are randomly assigned. This implies that there is no need to explicitly incorporate an alternative mediator in the analysis, for any kind of post-treatment confounding (observed or unobserved) is allowed to exist in the natural causal process to

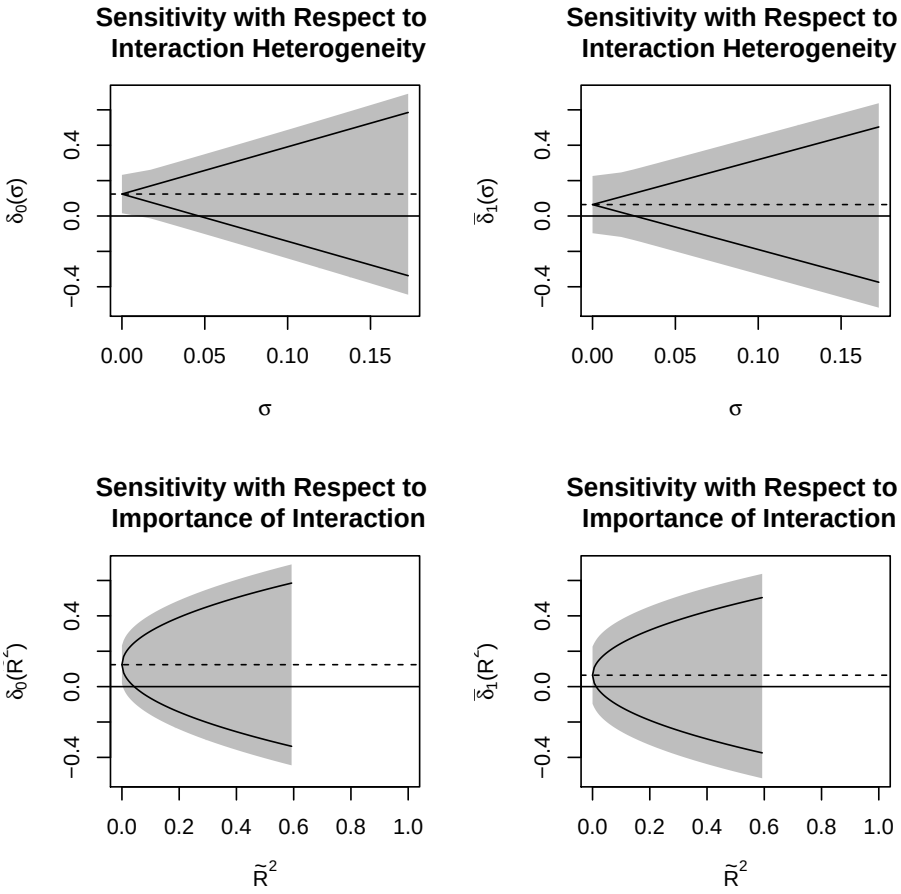

Figure 6: Graphical summary of sensitivity analysis using the `multimed` function. Results as a function of σ and $\widetilde{R}^2$.

identify the ACME under the parallel design using the proposed framework.

To apply the framework for the parallel design, one can use the `multimed` function with slightly modified inputs. First, the `med.alt` is no longer needed because the estimation framework is agnostic about what particular alternative mechanisms are confounding the mediator-outcome relationship. Second, one need to supply an additional variable (`experiment`) indicating whether units are assigned to the experiment with (`1`) or without (`0`) mediator manipulations. Finally, the `design` argument must be set to `"parallel"`, as opposed to the default value of `"single"`. For illustration, we again use the simulated data introduced in Section 5.2 and apply the varying coefficient linear structural equations framework.

```
R> m.med.para <- multimed(outcome = "out", med.main = "med", treat = "ttt",
+                         experiment = "manip", design = "parallel",
+                         data = boundsdata, sims = 100)
R> summary(m.med.para)

Causal Mediation Analysis with Confounding by an Alternative Mechanism
```

```
Estimates under the Homogeneous Interaction Assumption:
               Estimate 95% CI Lower 95% CI Upper
ACME (treated)    0.362        0.235         0.49
ACME (control)    0.307        0.188         0.43
ACME (average)    0.322        0.204         0.44
Total Effect      0.206        0.120         0.28

Sensitivity Analysis:
Values of the sensitivity parameters at which ACME first crosses zero:
               sigma(bounds) sigma(CI) R2s(bounds) R2s(CI) R2t(bounds) R2t(CI)
ACME (treated)         0.779     0.543       0.370   0.180       0.344    0.17
ACME (control)         0.627     0.425       0.240   0.110       0.223    0.10
ACME (average)         0.665     0.462       0.270   0.130       0.251    0.12
```

The `plot` function can also be used in the same manner as in the single experiment case. The key differences between the above analysis and Section 5.2 are threefold. First, the point estimates in the first summary table here only rely on the homogeneous interaction assumption, not the stronger assumption of no interaction. Second, however, the estimates here depend on the additional assumption of additivity, which is embodied in the varying coefficient structural equations model in Equations 8 and 9. The additivity assumption may not be plausible in some applications and needs to be carefully examined. Finally, the second summary table shows the result of the sensitivity analysis where the homogeneous interaction assumption is gradually relaxed until an arbitrary interaction heterogeneity is allowed. This may be preferred to the nonparametric bounds approach in Section 5.2, which offers less nuanced information about how robust the point estimates are to the violation of the identification assumption.

7. Causal mediation analysis with treatment noncompliance

A common complication in randomized controlled trials is treatment noncompliance. That is, experimental subjects may not follow the assigned treatment and instead choose to take another treatment. This poses a serious challenge to causal analysis in randomized experiments because, even though the *assigned* treatment is randomized by the researcher, the *actual* treatment is selected by the subjects themselves, quite possibly depending on certain characteristics unobserved to the researcher. In this section, we provide an overview of the method developed by Yamamoto (2013) to cope with the challenge of treatment noncompliance in the context of causal mediation analysis. The estimation method is implemented by the `ivmediate` function in our package, as discussed below.

7.1. The methodology

Analysis of causal mediation in the presence of treatment noncompliance requires additional notation and assumptions. Let $Z_i \in \{0, 1\}$ denote the binary indicator of treatment assignment, or encouragement, for unit i. Then, we use $T_i(z) \in \{0, 1\}$ to denote the potential treatment which unit i would actually receive when the unit were assigned to the treatment ($z = 1$) or control ($z = 0$) condition. The observed treatment for unit i can then be written

as $T_i = T_i(Z_i)$. Following the standard practice in the analysis of treatment noncompliance (see Angrist, Imbens, and Rubin 1996), we assume that the treatment assignment itself does not directly affect either the mediator or the outcome. Under these *exclusion restrictions*, we can write the potential mediator and outcome as, respectively, $M_i(t)$ and $Y_i(t, m)$. Likewise, the observed mediator and outcome can respectively be expressed as $M_i = M_i(T_i)$ and $Y_i(T_i, M_i(T_i))$. Another standard assumption we make is the *monotonicity* of treatment reception. That is, we assume that there is no unit in the population who would only take the treatment if assigned to the control condition. This assumption is thus often called the "no defier" assumption and can be written in our notation as $T_i(0) \leq T_i(1)$ for any i.

The final and key assumption is *local sequential ignorability*, which can be formally written as follows.

Assumption 2 (Local Sequential Ignorability; Yamamoto 2013)

$$\{Y_i(t', m), M_i(t), T_i(z)\} \perp\!\!\!\perp Z_i \mid X_i = x, \tag{11}$$

$$Y_i(t', m) \perp\!\!\!\perp M_i(t) \mid T_i = t, T_i(0) = 0, T_i(1) = 1, X_i = x, \tag{12}$$

for $t = 0, 1$, $z = 0, 1$, and all x and m in the support of X_i and M_i, respectively.

This assumption is closely related to but slightly weaker than the *global* sequential ignorability introduced earlier (Assumption 1). Equation 11 implies that the treatment assignment, Z_i, must be either randomized or can be regarded to be "as-if randomized" after conditioning on a set of observed pre-encouragement covariates, X_i. Equation 12, on the other hand, requires that the observed mediator in each treatment condition be as-if randomized among the *compliers*, i.e., those units whose actual treatment would always agree with their assigned treatment ($T_i(0) = 0$ and $T_i(1) = 1$). In other words, Assumption 2 implies that sequential ignorability must hold locally among the compliers.

In the context of treatment noncompliance, researchers typically focus on the estimation of the intent-to-treat (ITT) effect and the local average treatment effect (LATE) among the compliers. The former quantity refers to the average causal effect of the treatment assignment itself (regardless of the actual treatment) on the outcome, whereas the latter represents the average effect of the actual treatment on the outcome among the compliers. Here, we consider the problem of decomposing each of these effects into the direct and indirect portions with respect to the mediator of interest. First, the ITT effect can be written as the sum of these two quantities.

Mediated intent-to-treat (ITT) effect:

$$\bar{\lambda}(z) \equiv \mathbb{E}[Y_i(T_i(z), M_i(T_i(1))) - Y_i(T_i(z), M_i(T_i(0)))]. \tag{13}$$

Unmediated ITT effect:

$$\bar{\mu}(z) \equiv \mathbb{E}[Y_i(T_i(1), M_i(T_i(z))) - Y_i(T_i(0), M_i(T_i(z)))]. \tag{14}$$

Second, the LATE can be decomposed into the following two quantities.

Local average causal mediation effect (LACME):

$$\bar{\phi}(t) \equiv \mathbb{E}[Y_i(t, M_i(1)) - Y_i(t, M_i(0)) \mid T_i(0) = 0, T_i(1) = 1]. \tag{15}$$

Local average natural direct effect (LANDE):

$$\bar{\psi}(t) \equiv \mathbb{E}[Y_i(1, M_i(t)) - Y_i(0, M_i(t)) \mid T_i(0) = 0, T_i(1) = 1]. \quad (16)$$

Yamamoto (2013) shows that the above four quantities (each defined for $z = 0, 1$ or $t = 0, 1$) can be nonparametrically identified under the set of assumptions introduced thus far in this section. More specifically, each of $\bar{\lambda}(z)$, $\bar{\mu}(z)$, $\bar{\phi}(t)$ and $\bar{\psi}(t)$ can be expressed in terms of the conditional expectations and distributions of the observed outcome, mediator and treatment variables. These effects of interest can therefore be estimated by fitting regression models to each of those conditionals (i.e., a model for Y_i given M_i, T_i, Z_i and X_i, a model for M_i given T_i, Z_i and X_i, and a model for T_i given Z_i and X_i) and calculating the sample analogues of the identified quantities. Uncertainty estimates can then be obtained via simulation-based methods. The `ivmediate` function in our package implements this procedure for a selection of models, as we illustrate with an empirical example in the next section.

7.2. Illustration

We illustrate the use of the `ivmediate` function through an analysis of data from the Job Search Intervention Study (JOBS II; see Vinokur, Price, and Schul 1995, for more information about the study). JOBS II was a randomized job training intervention for unemployed workers which aimed at increasing the prospect of reemployment and improving mental health of the job seekers involved in the study. A random subsample of the participants were offered to receive the treatment of job-skills workshops which covered skills for job search and coping with stress, while the remaining participants were assigned to the control group and sent a booklet containing job-search tips. Despite the random assignment of the treatment conditions, some participants failed to comply with their assigned treatment status. Namely, 39% of those who were offered to participate in job-skills workshops actually did not enroll. None of the workers in the control group, on the other hand, participated in workshops, so noncompliance in this study was strictly one-sided. Several outcome measures were taken after the completion of the program via a survey. Here, we focus on the question of whether participation in the workshops improved the mental health of the unemployed workers (measured with a continuous scale based on the Hopkins Symptom Checklist) by increasing their self-confidence in job search ability (measured by a dichotomous indicator).

The relevant portion of the JOBS II data is included as part of the **mediation** package.

```
R> data("jobs", package = "mediation")
```

We begin by estimating three regression models. First, we model the actual treatment status (`comply`) conditional on the assigned treatment (`treat`) and observed pre-encouragement covariates (`sex`, `age`, `marital`, `nonwhite`, `educ`, and `income`). Here we postulate a linear probability model for the probability of actually participating in the job-skills workshops.

```
R> a <- lm(comply ~ treat + sex + age + marital + nonwhite + educ + income,
+          data = jobs)
```

Next, we model the mediator (`job_dich`) and outcome (`depress2`) as functions of causally precedent variables. That is, we fit a logit model for the dichotomous mediator conditional on the actual treatment, assigned treatment, and observed pre-encouragement covariates, and

a linear regression model for the outcome as a function of the mediator, actual treatment, assigned treatment, and pre-encouragement covariates.

```
R> b <- glm(job_dich ~ comply + treat + sex + age + marital +
+           nonwhite + educ + income, data = jobs, family = binomial)
R> c <- lm(depress2 ~ job_dich * (comply + treat) + sex + age + marital +
+           nonwhite + educ + income, data = jobs)
```

Generally, it is wise to include an interaction term between the actual and assigned treatment in these models in order to allow for the regression functions to vary across treatment conditions. Here, however, the interaction term is not needed because noncompliance is strictly one-sided (i.e., there is no observation for which `comply` equals `1` and `treat` equals `0`). These three models can in theory be of any form, as in the case of estimating the ACME via the `mediate` function. However, the current version of the `ivmediate` function only supports binary outcome models (fitted via `glm` with `family = binomial`) and linear models (fitted via `lm`).

After fitting the three models, the LACME and LANDE can be easily estimated via the `ivmediate` function, which takes those three fitted model objects as main inputs.[9]

```
R> out <- ivmediate(a, b, c, sims = 100, boot = FALSE,
+                   enc = "treat", treat = "comply", mediator = "job_dich")
```

In the `ivmediate` function, the analyst must specify the names of the assigned treatment, actual treatment, and mediator (as they appear in the data frames used to fit the three models) via the `enc`, `treat`, and `mediator` arguments, respectively. The analyst can also set the number of simulations used for the construction of confidence intervals (`sims`) as well as whether the nonparametric bootstrap should be used for the confidence intervals (`boot`; if `FALSE`, the quasi-Bayesian Monte Carlo method will be used). Regardless of this choice, only the intervals for the confidence levels specified by the `conf.level` argument (defaults to `.95`) will be calculated and retained in the output object (unless the `long` option is set to `TRUE`, in which case the entire set of simulation draws will be available).

An important remark is required on the computational demand of the `ivmediate` function. The estimation method of Yamamoto (2013) involves numerical integration over the support of the mediator for each observation in each simulation iteration. This implies that, if the mediator is continuous (i.e., if `model.m` is an 'lm; object) and the model contains any pre-encouragement covariate, the calculation of confidence intervals via `ivmediate` is extremely time-consuming. To ameliorate the situation, we have implemented a parallel execution of the simulation routine across multiple CPU cores. To utilize this function, the analyst should set the `multicore` option to `TRUE` and `mc.cores` to the desired number of cores available on his or her machine. Using N cores will approximately decrease the total computation time by the factor of N. This option, unfortunately, is implemented via `mclapply` and therefore not currently available on Windows machines.

The output of `ivmediate` can be summarized using the usual `summary` function.

```
R> summary(out)
```

[9] Currently, `ivmediate` only estimates the complier average effects and is not capable of estimating the mediated and unmediated ITT effects. This limitation will be addressed in a future update.

```
Causal Mediation Analysis with Treatment Noncompliance

Confidence Intervals Based on Quasi-Bayesian Monte Carlo

                 Estimate 95% CI Lower 95% CI Upper
LACME (control) -0.036963    -0.089771      -0.0035
LACME (treated) -0.042931    -0.086510      -0.0137
LANDE (control) -0.045669    -0.152165       0.1133
LANDE (treated) -0.051637    -0.157129       0.1189
LATE            -0.088600    -0.194514       0.0596

Sample Size Used: 899


Simulations: 100
```

The resulting summary table shows the point estimates of the LACME for the control and treatment baseline conditions ($\bar{\phi}(0)$ and $\bar{\phi}(1)$), the LANDE for the control and treatment baselines ($\bar{\psi}(0)$ and $\bar{\psi}(1)$), and the total LATE for the compliers in the first column, as well as their confidence intervals in the next two columns (unless the `ci` argument in `ivmediate` is set to `FALSE`, which makes the evaluation much faster). The confidence level is by default set at the first level used in the original `ivmediate` run, but can be changed to any level used via the `conf.level` option. Here, we observe that the indirect effect of job-skills workshop on depressive symptoms via increase in job-search self-efficacy is on average negative and barely statistically significant among the compliers, although the overall negative effect of treatment on depression among the compliers misses the conventional level of statistical significance.

The results can also be represented graphically via the `plot` function.

```
R> plot(out)
```

Again, the plotted confidence level can be changed via the `conf.level` option to any of the levels used in the original `ivmediate` call. The `effect.type` option can be used to specify which of the estimated quantities will be plotted (the default plots everything). Most of the standard graphical parameters can be set in the usual manner.

8. Concluding remarks

In this paper, we described the functionalities of the **mediation** package, which allows applied researchers to conduct causal mediation analysis in a variety of settings. The package implements a general algorithm for estimating causal mediation effects with a variety of statistical models. Since the causal mediation analysis under the standard research design requires a strong and untestable assumption, we recommend sensitivity analysis which is also implemented in our package. Finally, this strong identification assumption can be relaxed by adopting alternative research designs and we show how to use our package to conduct causal mediation analysis under those new designs. The literature on causal mediation analysis is fast growing, and we expect new methods to be developed in the coming years. We hope

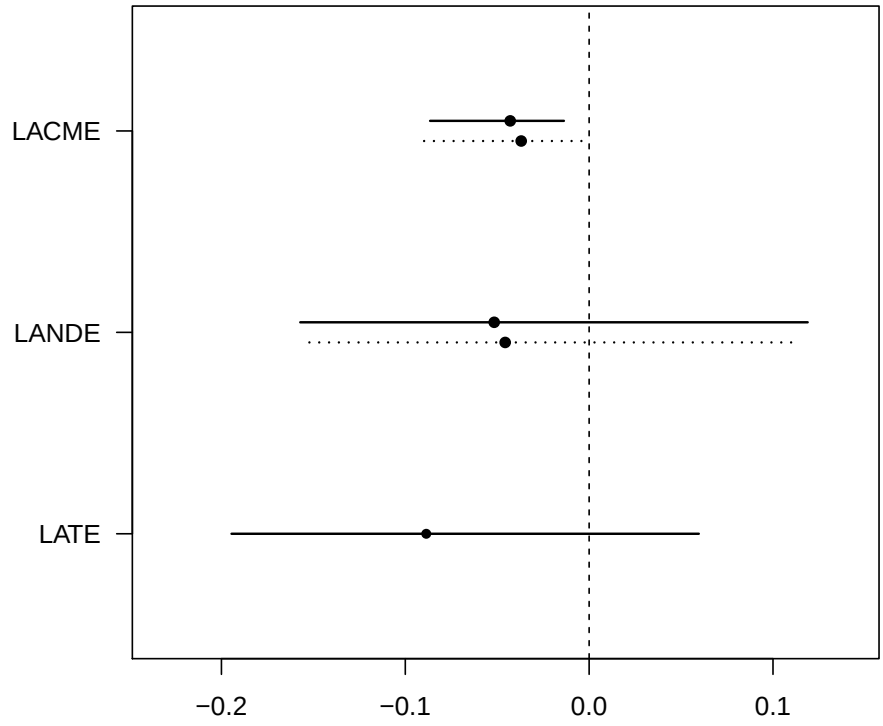

Figure 7: Graphical display of results from the `ivmediate` function.

that the **mediation** package can serve as a platform to which other researchers can add new methods.

References

Angrist JD, Imbens GW, Rubin DB (1996). "Identification of Causal Effects Using Instrumental Variables." Journal of the American Statistical Association, **91**(434), 444–455.

Baron RM, Kenny DA (1986). "The Moderator-Mediator Variable Distinction in Social Psychological Research: Conceptual, Strategic, and Statistical Considerations." Journal of Personality and Social Psychology, **51**(6), 1173–1182.

Bates D, Maechler M, Bolker B, Walker S (2014). **lme4**: Linear Mixed-Effects Models using **Eigen** and S4. R package version 1.1-7, URL http://CRAN.R-project.org/package=lme4.

Brader T, Valentino NA, Suhat E (2008). "What Triggers Public Opposition to Immigration? Anxiety, Group Cues, and Immigration." American Journal of Political Science, **52**(4), 959–978.

DiCiccio TJ, Efron B (1996). "Bootstrap Confidence Intervals." Statistical Science, **11**(3), 189–228.

Haavelmo T (1943). "The Statistical Implications of a System of Simultaneous Equations." Econometrica, **11**(1), 1–12.

Hicks R, Tingley D (2011). "Causal Mediation Analysis." Stata Journal, **11**(4), 609–615.

Imai K, Keele L, Tingley D (2010a). "A General Approach to Causal Mediation Analysis." Psychological Methods, **15**(4), 309–334.

Imai K, Keele L, Tingley D, Yamamoto T (2010b). "Causal Mediation Analysis Using R." In HD Vinod (ed.), Advances in Social Science Research Using R, Lecture Notes in Statistics, pp. 129–154. Springer-Verlag, New York.

Imai K, Keele L, Tingley D, Yamamoto T (2011). "Unpacking the Black Box: Learning about Causal Mechanisms from Experimental and Observational Studies." American Political Science Review, **105**(4), 765–789. URL http://imai.princeton.edu/research/mediationP.html.

Imai K, Keele L, Tingley D, Yamamoto T (2014). "Commentary: Practical Implications of Theoretical Results for Causal Mediation Analysis." Psychological Methods. Forthcoming.

Imai K, Keele L, Yamamoto T (2010c). "Identification, Inference, and Sensitivity Analysis for Causal Mediation Effects." Statistical Science, **25**(1), 51–71.

Imai K, Tingley D, Yamamoto T (2013). "Experimental Designs for Identifying Causal Mechanisms." Journal of the Royal Statistical Society A, **176**(1), 5–51.

Imai K, Yamamoto T (2013). "Identification and Sensitivity Analysis for Multiple Causal Mechanisms: Revisiting Evidence from Framing Experiments." Political Analysis, **21**(2), 141–171.

Krull JL, MacKinnon DP (2001). "Multilevel Modeling of Individual and Group Level Mediated Effects." Multivariate Behavioral Research, **36**(2), 249–277.

MacKinnon D (2008). Introduction to Statistical Mediation Analysis. Routledge, New York.

Pearl J (2001). "Direct and Indirect Effects." In Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence, pp. 411–420. Morgan Kaufmann, San Francisco.

Pearl J (2014). "Interpretation and Identification of Causal Mediation." Psychological Methods. doi:10.1037/a0036434. Forthcoming.

Preacher KJ, Hayes AF (2008). "Asymptotic and Resampling Strategies for Assessing and Comparing Indirect Effects in Multiple Mediator Models." Behavior Research Methods, **40**(3), 879–891.

R Core Team (2014). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Robins JM, Greenland S (1992). "Identifiability and Exchangeability for Direct and Indirect Effects." Epidemiology, **3**(2), 143–155.

Shadish WR, Cook TD, Campbell DT (2001). Experimental and Quasi-Experimental Designs for Generalized Causal Inference. Houghton Mifflin, Boston.

StataCorp (2013). Stata Data Analysis Statistical Software: Release 12. StataCorp LP, College Station, TX. URL http://www.stata.com/.

Tingley D, Yamamoto T, Hirose K, Keele L, Imai K (2013). **mediation**: R Package for Causal Mediation Analysis. R package version 4.4.2, URL http://CRAN.R-project.org/package=mediation.

Vinokur AD, Price RH, Schul Y (1995). "Impact of the JOBS Intervention on Unemployed Workers Varying in Risk for Depression." American Journal of Community Psychology, **23**(1), 39–74.

Yamamoto T (2013). "Identification and Estimation of Causal Mediation Effects with Treatment Noncompliance." Unpublished Manuscript.

Zeileis A (2006). "Object-Oriented Computation of Sandwich Estimators." Journal of Statistical Software, **16**(9), 1–16. URL http://www.jstatsoft.org/v16/i09/.

Zhang Z, Zyphur MJ, Preacher KJ (2009). "Testing Multilevel Mediation Using Hierarchical Linear Models." Organizational Research Methods, **12**(4), 695–719.

Affiliation:

Dustin Tingley
Department of Government
Harvard University
1737 Cambridge St
Cambridge, MA, United States of America
E-mail: dtingley@gov.harvard.edu
URL: http://scholar.harvard.edu/dtingley

Teppei Yamamoto
Department of Political Science
Massachusetts Institute of Technology
77 Massachusetts Avenue, E53-463
Cambridge, MA, United States of America
E-mail: teppei@mit.edu
URL: http://web.mit.edu/teppei/www/

Kentaro Hirose, Kosuke Imai
Department of Politics
Princeton University
Princeton, NJ, United States of America
E-mail: hirose@princeton.edu, kimai@princeton.edu
URL: http://imai.princeton.edu.

Luke Keele
Department of Political Science
Pennsylvania State University
211 Pond Lab
University Park, PA, United States of America
Email: ljk20@psu.edu
URL: http://www.personal.psu.edu/ljk20