Robust Mediation Analysis: The R Package robmed

Andreas Alfons
Erasmus University Rotterdam

Nüfer Y. Ateş
Sabancı University

Patrick J.F. Groenen
Erasmus University Rotterdam

Abstract

Mediation analysis is one of the most widely used statistical techniques in the social, behavioral, and medical sciences. Mediation models allow to study how an independent variable affects a dependent variable indirectly through one or more intervening variables, which are called mediators. The analysis is often carried out via a series of linear regressions, in which case the indirect effects can be computed as products of coefficients from those regressions. Statistical significance of the indirect effects is typically assessed via a bootstrap test based on ordinary least-squares estimates. However, this test is sensitive to outliers or other deviations from normality assumptions, which poses a serious threat to empirical testing of theory about mediation mechanisms. The R package **robmed** implements a robust procedure for mediation analysis based on the fast-and-robust bootstrap methodology for robust regression estimators, which yields reliable results even when the data deviate from the usual normality assumptions. Various other procedures for mediation analysis are included in package **robmed** as well. Moreover, **robmed** introduces a new formula interface that allows to specify mediation models with a single formula, and provides various plots for diagnostics or visual representation of the results.

Keywords: mediation analysis, robust statistics, bootstrap, R.

1. Introduction

In the social, behavioral, and medical sciences, *mediation analysis* is a popular statistical technique for studying how an *independent variable* affects a *dependent variable* indirectly through an intervening variable called a *mediator*. For instance, Erreygers, Vandebosch, Vranjes, Baillien, and De Witte (2018) find that poor sleep quality in adolescents explains cyberbullying through anger, and Gaudiano, Herbert, and Hayes (2010) report that the believability of hallucinations after treatment for psychotic disorders mediates the relationship between the type of treatment and distress after treatment. Figure 1 shows a diagram of the mediation model in its simplest form, which is given by the equations

$$M = i_1 + aX + e_1, \tag{1}$$
$$Y = i_2 + bM + cX + e_2, \tag{2}$$
$$Y = i_3 + c'X + e_3, \tag{3}$$

where Y denotes the dependent variable, X the independent variable, M the hypothesized mediator, i_1, i_2, i_3, a, b, c, and c' are regression coefficients to be estimated, and e_1, e_2, and e_3 are random error terms. The coefficients c and c' are called the *direct effect* and *total effect*, respectively, of X on Y. The product of coefficients ab is called the *indirect effect* of

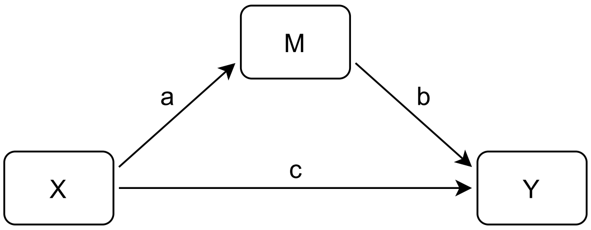

Figure 1: Diagram visualizing a simple mediation model.

X on Y and constitutes the main parameter of interest in mediation analysis. Under the usual assumption of independent and normally distributed error terms e_1, e_2 and e_3, it holds that $c' = ab + c$ (e.g., MacKinnon, Warsi, and Dwyer 1995), and the same holds for the corresponding ordinary least-squares (OLS) estimates.

Mediation analysis goes back to Judd and Kenny (1981) and Baron and Kenny (1986), however their stepwise approach has been superseded by approaches that focus on the indirect effect. Sobel (1982) proposed a test for the indirect effect that assumes a normal distribution of the corresponding estimator, which is an unrealistic assumption for a product of coefficients. Bollen and Stine (1990), Shrout and Bolger (2002), MacKinnon, Lockwood, and Williams (2004), and Preacher and Hayes (2004, 2008) therefore advocate for a bootstrap test based on OLS regressions, which is the most frequently applied method for mediation analysis according to literature reviews of Wood, Goodman, Beckmann, and Cook (2008) and Alfons, Ateş, and Groenen (2021). More recently, several authors have emphasized that outliers or deviations from normality assumptions are detrimental to the reliability and validity of mediation analysis, and introduced more robust procedures. Zu and Yuan (2010) propose a bootstrap test after an initial data cleaning step, whereas Yuan and MacKinnon (2014) suggest a bootstrap test based on median regressions. Alfons *et al.* (2021) combine the robust MM-estimator of regression (Yohai 1987) with the the fast-and-robust bootstrap (Salibián-Barrera and Zamar 2002; Salibián-Barrera and Van Aelst 2008), and demonstrate that this procedure outperforms the aforementioned approaches for a wide range of error distributions (with different levels of skewness and kurtosis) and outlier configurations.

Various software packages are available for mediation analysis. The macro `INDIRECT` (Preacher and Hayes 2004, 2008) and its successor `PROCESS` (Hayes 2018) for SPSS (IBM Corp. 2021) and SAS (SAS Institute Inc. 2020) implement the bootstrap test based on OLS regressions. For the statistical computing environment R (R Core Team 2022), the general purpose packages **psych** (Revelle 2021) and **MBESS** (Kelley 2022) for statistical analysis in the behavioral sciences also provide functionality for a bootstrap test in mediation analysis. Package **WRS2** (Mair and Wilcox 2020) is a collection of robust statistical methods, which offers mediation analysis via the bootstrap test after data cleaning proposed by Zu and Yuan (2010). Other packages concentrate on mediation analysis or specific aspects thereof. Package **mediation** (Tingley, Yamamoto, Hirose, Keele, and Imai 2014) is focused on causal mediation analysis in a potential outcome framework, and package **medflex** (Steen, Loeys, Moerkerke, and Vansteelandt 2017) implements recent developments in mediation analysis embedded within the counterfactual framework. Bayesian multilevel mediation models can be estimated with package **bmlm** (Vuorre 2021), while package **mma** (Yu and Li 2021) offers functionality for general multiple mediation analysis with continuous or binary/categorical variables. In addition, general purpose software for structural equation modeling such as Mplus (Muthén and Muthén 2017) or the R packages **sem** (Fox, Nie, and Byrnes 2022) and **lavaan** (Rosseel 2012) can be used for mediation analysis. The former also allows for maximum likelihood estimation with skew-normal, t, or skew-t error distributions (Asparouhov and Muthén 2016).

Despite the growing number of R packages that address mediation analysis, there are no common interfaces or class structures. Instead, each package uses its own way of specifying mediation models and storing the results. Additionally, only package **WRS2** contains some functionality for robust mediation analysis.

Package **robmed** (Alfons 2022b) aims to address these issues. Its main functionality is the robust bootstrap procedure proposed in Alfons *et al.* (2021), which is highly robust to outliers and other deviations from normality assumptions. Furthermore, **robmed** implements various other methods of estimating mediation models, as well as different tests for the indirect effects. All implemented methods share the same function interface and a clear class structure. In addition, **robmed** introduces a simple formula interface for specifying mediation models, and provides several plots for diagnostics or visualization of the results from mediation analysis.

Package **robmed** is available on CRAN (the Comprehensive R Archive Network, `https://CRAN.R-project.org/`) and can be installed from the R console with the following command:

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

The rest of the paper is organized as follows. Section 2 discusses various extensions of the simple mediation model, as well as the implemented methodology for estimation and testing. Implementation details are provided in Section 3, while Section 4 illustrates the use of package **robmed** with code examples. The final Section 5 concludes.

2. Methodology

We first provide overviews of the mediation models and estimation techniques supported by package **robmed** in Sections 2.1 and 2.2, respectively. Section 2.3 then gives technical details of the robust bootstrap procedure of Alfons *et al.* (2021).

2.1. Extensions of the simple mediation model

The simple mediation model (1)-(3) can easily be extended in various ways, for instance with (i) multiple parallel mediators, (ii) multiple serial mediators, and (iii) multiple independent variables to be mediated. All those extensions may include additional control variables (covariates) as well.

Parallel multiple mediator model

In the *parallel multiple mediator model*, an independent variable X is hypothesized to influence a dependent variable Y through multiple mediators $M_1, \dots, M_k$, while the mediator variables do not influence each other. A diagram of the model is displayed in Figure 2, and the

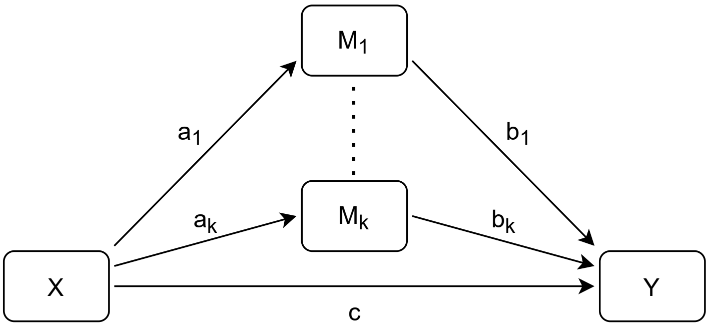

Figure 2: Diagram visualizing a parallel multiple mediator model.

corresponding regression equations are

$$M_j = i_j + a_j X + e_j, \qquad j = 1, \dots, k, \tag{4}$$
$$Y = i_{k+1} + b_1 M_1 + \dots + b_k M_k + cX + e_{k+1}, \tag{5}$$
$$Y = i_{k+2} + c' X + e_{k+2}, \tag{6}$$

where $i_1, \dots, i_{k+2}$, $a_1, \dots, a_k$, $b_1, \dots, b_k$, c, and c' are regression coefficients to be estimated, and $e_1, \dots, e_{k+2}$ are random error terms. With the usual assumptions of independent and normally distributed error terms, we now have that $c' = \sum_{j=1}^{k} a_j b_j + c$. The main parameters of interest are the individual indirect effects $a_1 b_1, \dots, a_k b_k$, and it can also be of interest to make pairwise comparisons between the individual indirect effects or their absolute values (e.g., Hayes 2018, Chapter 5.1) if the hypothesized mediators are scaled similarly.

Serial multiple mediator model

A distinctive feature of the *serial multiple mediator model* is that the hypothesized mediators $M_1, \dots, M_k$ may influence each other in a sequential manner, unlike the parallel multiple mediator model in which the mediators do not affect one another. Figure 3 contains a diagram of the model with two serial mediators, while the model in its general form is given by

$$\begin{aligned} M_1 &= i_1 + a_1 X + e_1, \\ M_2 &= i_2 + d_{21} M_1 + a_2 X + e_2, \\ &\vdots \\ M_k &= i_k + d_{k1} M_1 + \dots + d_{k,k-1} M_{k-1} + a_k X + e_k, \end{aligned} \tag{7}$$
$$Y = i_{k+1} + b_1 M_1 + \dots + b_k M_k + cX + e_{k+1}, \tag{8}$$
$$Y = i_{k+2} + c' X + e_{k+2}, \tag{9}$$

where $i_1, \dots, i_{k+2}$, $a_1, \dots, a_k$, $d_{j1}, \dots, d_{j,j-1}$, $j = 2, \dots, k$, $b_1, \dots, b_k$, c, and c' are regression coefficients to be estimated, and $e_1, \dots, e_{k+2}$ are random error terms. It is easy to see that the serial multiple mediator model quickly grows in complexity with increasing number of mediators due to the combinatorial increase in indirect paths through the mediators (the number of indirect paths is given by $\sum_{j=1}^{k} \binom{k}{j}$ for k serial mediators). In package **robmed**, it is therefore only implemented for two and three mediators to maintain a focus on easily

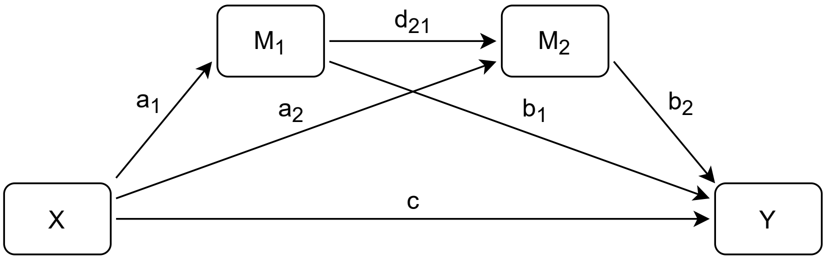

Figure 3: Diagram visualizing a serial multiple mediator model with two mediators.

interpretable models. Here, we only discuss the model for two serial mediators, and we refer to Hayes (2018, p.169–171) for a diagram and a description of the various indirect effects in the case of three serial mediators.

For two serial mediators, the three indirect effects $a_1 b_1$ ($X \to M_1 \to Y$), $a_2 b_2$ ($X \to M_2 \to Y$), and $a_1 d_{21} b_2$ ($X \to M_1 \to M_2 \to Y$) are the main parameters of interest. However, not all pairwise comparisons of the indirect effects may be meaningful (even if the mediators are scaled similarly), as $a_1 d_{21} b_2$ can be expected to be different in magnitude from $a_1 b_1$ and $a_2 b_2$. Finally, we have that $c' = a_1 b_1 + a_2 b_2 + a_1 d_{21} b_2 + c$ under the usual assumptions of independent and normally distributed error terms.

Multiple independent variables to be mediated

Instead of having multiple mediators, one can also allow for *multiple independent variables* $X_1, \dots, X_l$ to influence the dependent variable Y through a hypothesized mediator M. The resulting model is visualized in Figure 4 and defined by the equations

$$M = i_1 + a_1 X_1 + \cdots + a_l X_l + e_1, \tag{10}$$

$$Y = i_2 + bM + c_1 X_1 + \cdots + c_l X_l + e_2, \tag{11}$$

$$Y = i_3 + c'_1 X_1 + \cdots + c'_l X_l + e_3, \tag{12}$$

where i_1, i_2, i_3, $a_1, \dots, a_l$, b, $c_1, \dots, c_l$, and $c'_1, \dots, c'_l$ are regression coefficients to be estimated, and e_1, e_2, and e_3 are random error terms. The indirect effects $a_1 b, \dots, a_l b$ are the main parameters of interest, and with the direct effects $c_1, \dots, c_l$ and total effects $c'_1, \dots, c'_l$, it holds that $c'_j = a_j b + c_j$, $j = 1, \dots, l$, under the usual independence and normality assumptions on the error terms. If the independent variables are on a comparable scale, it can also be of interest to make pairwise comparisons between the indirect effects or their absolute values.

This model is commonly used when the hypothesized mediator is the main (explanatory) variable of interest and its antecedents are being studied. Furthermore, an important special case of this model occurs when a categorical independent variable is represented by a group of dummy variables.

Control variables

In many study designs, it is necessary to isolate the effects of the independent variables of interest from other factors. For this purpose, *control variables* should be added to all regression equations of a mediation model. This means that there is no intrinsic difference

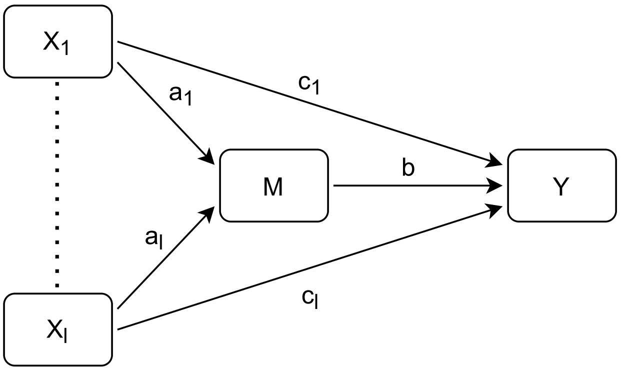

Figure 4: Diagram visualizing a mediation model with multiple independent variables.

between independent variables of interest and control variables in terms of the model or its estimation. The difference is purely conceptual in nature: for the control variables, the estimates of the direct and indirect paths are not of particular interest to the researcher. Package **robmed** therefore allows to specify control variables separately from the independent variables of interest. Only for the latter, results for the indirect effects are included in the output.

While we omitted control variables from the above equations and diagrams for notational simplicity, package **robmed** supports additional control variables in all implemented models.

More complex models

The models described above do not exist in isolation and some of them can be combined. For instance, **robmed** supports parallel and serial multiple mediator models with multiple independent variables of interest. Other variations of the mediation model, such as the moderated mediation and mediated moderation models (e.g., Muller, Judd, and Yzerbyt 2005) are out of scope for this paper. They are not yet implemented in package **robmed** but we aim to add support in future versions.

2.2. Overview of implemented methods

While package **robmed** is focused on the fast-and-robust bootstrap procedure for mediation analysis introduced by Alfons *et al.* (2021), various other methods are available as well. Tables 1 and 2 provide an overview of the available methods together with the corresponding argument values to use in function `test_mediation()`, which implements mediation analysis in **robmed**.

A bootstrap test is considered state-of-the art for mediation analysis, with many authors advocating to use the bootstrap with ordinary least-squares (OLS) estimation of the coefficients in the mediation model (Bollen and Stine 1990; Shrout and Bolger 2002; MacKinnon *et al.* 2004; Preacher and Hayes 2004, 2008). However, a bootstrap test can easily be applied to other methods of estimation. For instance, the mediation model can also be estimated via regressions with more flexible error distributions such as the skew-normal, t, or skew-t distributions (see Azzalini and Arellano-Valle 2013, for maximum likelihood estimation of such

test	method	robust	family	Description	Simple mediation	Parallel mediators	Serial mediators	Multiple independent variables of interest	Control variables
`"boot"`	`"regression"`	`"MM"` or `TRUE`		MM-regression (Yohai 1987) and the fast-and-robust bootstrap (Salibián-Barrera and Zamar 2002) are used to construct a confidence interval (Alfons *et al.* 2021).	✓	✓	✓	✓	✓
`"boot"`	`"regression"`	`"median"`		A bootstrap confidence interval is computed based on median regressions (Yuan and MacKinnon 2014).	✓	✓	✓	✓	✓
`"boot"`	`"regression"`	`FALSE`	`"gaussian"`	A bootstrap confidence interval is computed based on OLS regressions (Bollen and Stine 1990; Shrout and Bolger 2002; MacKinnon *et al.* 2004; Preacher and Hayes 2004, 2008).	✓	✓	✓	✓	✓
`"boot"`	`"regression"`	`FALSE`	`"select"`	Regression models with normal, skew-normal, t, or skew-t error distributions are estimated (Azzalini and Arellano-Valle 2013), and a bootstrap confidence interval is computed. The best fitting error distribution is selected via BIC.	✓	✓	✓	✓	✓
`"boot"`	`"covariance"`	`TRUE`		Following multivariate winsorization, a bootstrap confidence interval is computed for coefficient estimation via the maximum likelihood estimator of the covariance matrix (Zu and Yuan 2010).	✓				
`"boot"`	`"covariance"`	`FALSE`		A bootstrap confidence interval is computed for coefficient estimation via the maximum likelihood estimator of the covariance matrix.	✓				

Table 1: Overview of bootstrap procedures for mediation analysis implemented in the R package **robmed**, and corresponding argument values to use in function `test_mediation()`.

test	method	robust	family	Description	Simple mediation	Parallel mediators	Serial mediators	Multiple independent variables of interest	Control variables
`"sobel"`	`"regression"`	`"MM"` or `TRUE`		A variation of the Sobel test based on coefficient estimation via MM-regressions.	✓				✓
`"sobel"`	`"regression"`	`"median"`		A variation of the Sobel test based on coefficient estimation via median regressions.	✓				✓
`"sobel"`	`"regression"`	`FALSE`	`"gaussian"`	A variation of the Sobel test based on coefficient estimation via OLS regressions.	✓				✓
`"sobel"`	`"regression"`	`FALSE`	`"select"`	A variation of the Sobel test in which regression models with normal, skew-normal, *t*, or skew-*t* error distributions are estimated, with the best fitting distribution being selected via BIC.	✓				✓
`"sobel"`	`"covariance"`	`TRUE`		Following multivariate winsorization, a variation of the Sobel test is applied based on coefficient estimation via the maximum likelihood estimator of the covariance matrix.	✓				
`"sobel"`	`"covariance"`	`FALSE`		A variation of the Sobel test based on coefficient estimation via the maximum likelihood estimator of the covariance matrix.	✓				

Table 2: Overview of variations of the Sobel test (Sobel 1982) for mediation analysis implemented in the R package **robmed**, and corresponding argument values to use in function `test_mediation()`. Those tests are included for benchmarking purposes and are not recommended for empirical analyses.

regression models). Note that a similar procedure for mediation analysis, but using structural equation modeling, has been suggested in Asparouhov and Muthén (2016). Package **robmed** goes a step further in that it allows to select the best fitting error distribution via the Bayesian information criterion (BIC) (Schwarz 1978). In addition, other robust methods for mediation analysis are implemented in **robmed**. Yuan and MacKinnon (2014) proposed a bootstrap test that replaces OLS estimation with median regression. Zu and Yuan (2010) proposed to first winsorize the data via a Huber M-estimator of the covariance matrix, and then to perform a bootstrap test on the cleaned data with coefficient estimation based on the maximum likelihood covariance matrix. A discussion of advantages and disadvantages of those approaches, as well as a comparison in extensive simulation studies, can be found in Alfons *et al.* (2021).

Besides bootstrap tests, variations of the Sobel test (Sobel 1982) are implemented in **robmed**. The Sobel test was originally proposed for maximum likelihood estimation of structural equation models, of which mediation models are a special case. It assumes a normal distribution of the indirect effect estimator and simplifies the calculation of the standard error by taking a first- or second-order approximation. This test has been criticized in the literature for its incorrect assumptions (e.g., MacKinnon, Lockwood, Hoffman, West, and Sheets 2002), and a bootstrap test is generally preferred. Nevertheless, the Sobel test can easily be generalized to other estimation methods, and it is implemented in **robmed** for all estimation procedures of the (simple) mediation model. We emphasize that the Sobel tests are implemented for comparisons in benchmarking experiments and that they are not recommended for empirical analyses.

2.3. Fast-and-robust bootstrap test for mediation analysis

The robust procedure of Alfons *et al.* (2021) follows the state-of-the-art bootstrap approach for testing mediation (Bollen and Stine 1990; Shrout and Bolger 2002; MacKinnon *et al.* 2004; Preacher and Hayes 2004, 2008), but it replaces OLS regressions with the robust MM-estimator of regression (Yohai 1987) and the standard bootstrap with the fast-and-robust bootstrap (Salibián-Barrera and Zamar 2002; Salibián-Barrera and Van Aelst 2008).

Robust regression

For a response variable Y, a $(p+1)$-dimensional random vector $\boldsymbol{X}$ in which the first component is fixed at 1, and a random error term $\varepsilon \sim N(0, \sigma^2)$, the linear regression model is given by

$$Y = \boldsymbol{X}^\top \boldsymbol{\beta} + \varepsilon.$$

Denoting the corresponding observations by $(y_i, \mathbf{x}_i^\top)^\top$, $i = 1, \dots, n$, the MM-estimate of regression with loss function ρ (Yohai 1987) is defined as

$$\hat{\boldsymbol{\beta}}_n = \operatorname*{argmin}_{\beta} \sum_{i=1}^{n} \rho \left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}_n} \right), \tag{13}$$

where $r_i(\boldsymbol{\beta}) = y_i - \mathbf{x}_i^\top \boldsymbol{\beta}$, $i = 1, \dots, n$, are the residuals, and $\hat{\sigma}_n$ is an initial estimate of the residual scale. We take $\hat{\sigma}_n$ from a highly robust but inefficient S-estimator of regression (Rousseeuw and Yohai 1984; Salibián-Barrera and Yohai 2006), and we use Tukey's bisquare loss function defined as

$$\rho(x) = \begin{cases} \dfrac{x^6}{6c^4} - \dfrac{x^4}{2c^2} + \dfrac{x^2}{2}, & \text{if } |x| \leq c \\ \dfrac{c^2}{6}, & \text{if } |x| > c, \end{cases} \tag{14}$$

where c is a tuning constant that determines the efficiency of the estimator. We set $c = 3.443689$, which results in 85% efficiency at the model with normally distributed errors. Taking the derivative of (13) and equating to $\mathbf{0}$ yields the system of equations

$$\sum_{i=1}^{n} \rho' \left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}_n} \right) \mathbf{x}_i = \mathbf{0}. \tag{15}$$

With weights

$$w_i = \frac{\rho'(r_i(\boldsymbol{\beta})/\hat{\sigma}_n)}{r_i(\boldsymbol{\beta})/\hat{\sigma}_n}, \qquad i = 1, \dots, n,$$

the system of equations in (15) can be rewritten as a weighted version of the normal equations:

$$\sum_{i=1}^{n} w_i r_i(\boldsymbol{\beta}) \mathbf{x}_i = \mathbf{0}. \tag{16}$$

Therefore, the MM-estimator can be seen as a weighted least-squares estimator with data-dependent weights. Those weights indicate how much each observation deviates, as an observation with a large residual (large deviation) receives a weight of 0 or close to 0, while an observation with a small residual (small deviation) receives a weight close to 1. The loss function from (14) and the resulting weight function are displayed in Figure 5, which also includes the loss function and weight function from OLS regression for comparison.

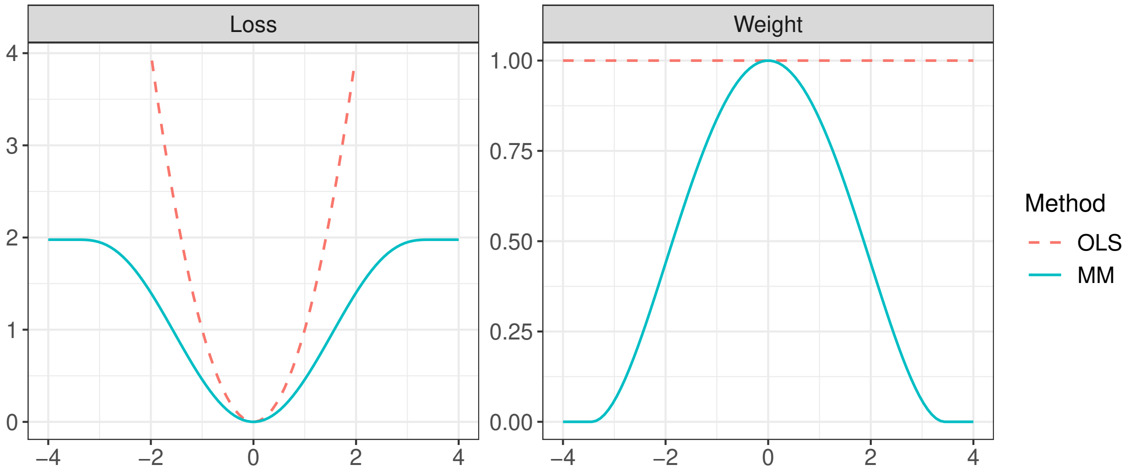

Figure 5: Loss function (*left*) and corresponding weight function (*right*) for OLS regression and the robust MM-estimator of regression.

Fast-and-robust bootstrap

Here we briefly present the main idea of the fast-and-robust bootstrap. For complete derivations, the reader is referred to Salibián-Barrera and Zamar (2002) and Salibián-Barrera and Van Aelst (2008). Note that the solution of (16) can be written as

$$\hat{\boldsymbol{\beta}}_n = \left(\sum_{i=1}^{n} w_i \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} w_i \mathbf{x}_i y_i.$$

For a bootstrap sample $(y_i^*, \mathbf{x}_i^{*\top})^\top$, $i = 1, \ldots, n$, one can compute $r_i^* = y_i^* - \mathbf{x}_i^{*\top} \hat{\boldsymbol{\beta}}_n$ and $w_i^* = \rho'(r_i^*/\hat{\sigma}_n)/(r_i^*/\hat{\sigma}_n)$ for $i = 1, \ldots, n$. It is important to note that $\hat{\boldsymbol{\beta}}_n$ and $\hat{\sigma}_n$ are computed in advance on the original sample such that the robustness weights w_i^* are inherited from the respective observations in the original sample. Then only a weighted least-squares fit is computed on the bootstrap sample to obtain

$$\hat{\boldsymbol{\beta}}_n^{\mathrm{WLS}} = \left(\sum_{i=1}^{n} w_i^* \mathbf{x}_i^* \mathbf{x}_i^{*\top} \right)^{-1} \sum_{i=1}^{n} w_i^* \mathbf{x}_i^* y_i^*. \tag{17}$$

However, using (17) for the bootstrap distribution would not capture all the variability in the MM-estimator, as the robustness weights are not recomputed on the bootstrap samples. Nevertheless, a linear correction of the coefficients can be applied to overcome this loss of variability. The correction matrix only needs to be computed once based on the original sample and is given by

$$\boldsymbol{K}_n = \left(\sum_{i=1}^{n} \rho''(r_i/\hat{\sigma}_n) \mathbf{x}_i \mathbf{x}_i^\top \right)^{-1} \sum_{i=1}^{n} w_i \mathbf{x}_i \mathbf{x}_i^\top.$$

Then the fast-and-robust bootstrap replicate on the given bootstrap sample is computed as

$$\hat{\boldsymbol{\beta}}_n^* = \hat{\boldsymbol{\beta}}_n + \boldsymbol{K}_n (\hat{\boldsymbol{\beta}}_n^{\mathrm{WLS}} - \hat{\boldsymbol{\beta}}_n). \tag{18}$$

Since the MM-estimator $\hat{\boldsymbol{\beta}}_n$ is consistent for $\boldsymbol{\beta}$ (Yohai 1987), $\sqrt{n}(\hat{\boldsymbol{\beta}}_n^* - \hat{\boldsymbol{\beta}}_n)$ has the same asymptotic distribution as $\sqrt{n}(\hat{\boldsymbol{\beta}}_n - \boldsymbol{\beta})$ (Salibián-Barrera and Van Aelst 2008; Salibián-Barrera and Zamar 2002).

Bootstrapping the indirect effects in mediation analysis

For simplicity, we focus on the indirect effect in the simple mediation model from (1)–(3). Similar calculations apply to the indirect effects in the mediation models described in Section 2.1. On each bootstrap sample, (18) is used to obtain estimates $\hat{a}$, $\hat{b}$, and $\hat{c}$ of the coefficients in (1) and (2), and therefore estimates $\hat{a}\hat{b}$ of the indirect effect. Note that we do not perform the regression corresponding to (3) and instead estimate the total effect by $\hat{c}' = \hat{a}\hat{b} + \hat{c}$. With the empirical distribution of the indirect effect over the bootstrap samples, we construct a percentile-based confidence interval. By default, we report a bias-corrected and accelerated confidence interval (Davison and Hinkley 1997). Furthermore, we advocate to report the mean over the bootstrap distribution as the final point estimates of the indirect effect.

3. Implementation

We introduce the formula interface for specifying mediation models in Section 3.1, and briefly discuss the main functions as well as the class structure of package **robmed** in Section 3.2.

3.1. Formula interface

The equations in the mediation model follow a specific structure regarding which variable is used as the response variable and which variables are the explanatory variables. Some R packages for mediation analysis, e.g., **mediation** (Tingley *et al.* 2014), require the user to specify one formula for each equation, which can be tedious and prone to mistakes, in particular for models with multiple mediators and multiple independent variables or control variables. Other packages, e.g., **psych** (Revelle 2021) or **MBESS** (Kelley 2022), do not provide a formula interface at all, despite formulas being the standard way of describing models in R.

For package **robmed**, we designed a formula interface that builds upon the standard formula interface in R, but allows to specify the mediation model with a single formula. As usual, the dependent variable is defined on the left hand side of the formula, and the independent variable is given on the right hand side. In addition, the functions `m()` and `covariates()` can be used on the right hand side to define the hypothesized mediators and any control variables, respectively. If multiple mediators are supplied, function `m()` provides the argument `.model`, which accepts the values `"parallel"` for parallel mediators (the default) and `"serial"` for serial mediators. The corresponding wrapper functions `parallel_m()` and `serial_m()` are available for convenience.

For example, a simple mediation model can be defined as follows (see also the case study in Section 4.1):

```
R> TeamCommitment ~ m(TaskConflict) + ValueDiversity
```

An example for a serial multiple mediator model is specified with the following formula (see also the case study in Section 4.2), where the serial mediators are listed in consecutive order from left to right:

```
R> TeamScore ~ serial_m(TaskConflict, TeamCommitment) + ValueDiversity
```

The formula specification for an example of a parallel multiple mediator model with control variables is given by (see also the case study in Section 4.3):

```
R> TeamPerformance ~ parallel_m(ProceduralJustice, InteractionalJustice) +
+    SharedLeadership + covariates(AgeDiversity, GenderDiversity)
```

Note that different variables within `m()`, `parallel_m()`, `serial_m()`, and `covariates()` are separated by commas.

3.2. Main functions and class structure

The two main functions of package **robmed** are `fit_mediation()`, which implements various methods for the *estimation* of a mediation model, and `test_mediation()`, which performs *statistical tests* on the indirect effects in the mediation model.

Function `fit_mediation()` is mainly intended to be used internally by `test_mediation()`, but it is also useful for a user who wants to compare different tests on the indirect effects for the same method of estimation, such that the estimation on the given sample only has to be performed once. It returns an object inheriting from class `"fit_mediation"`. The currently available subclasses are `"reg_fit_mediation"` if the mediation model was estimated via a series of regressions, and `"cov_fit_mediation"` if the model was estimated based on the covariance matrix of the involved variables.

We expect most users to find it more convenient to use `test_mediation()` directly in order to perform model estimation and testing the indirect effects with one function call. See Tables 1 and 2 for an overview of which argument values to use in `test_mediation()` for the various available methods. Furthermore, function `robmed()` is a wrapper function for the fast-and-robust bootstrap test of Alfons *et al.* (2021). The results are returned as an object inheriting from class `"test_mediation"`. The currently available subclasses are `"boot_test_mediation"` for bootstrap tests, and `"sobel_test_mediation` for tests based on the normal approximation of Sobel (1982). Among other information, the component `fit` stores the estimation results as an object inheriting from class `"fit_mediation"`. Objects of class `"boot_test_mediation"` also contain a component `reps`, which stores the bootstrap replicates as an object of class `"boot"`, as returned by function `boot()` from package **boot** (Canty and Ripley 2021). It should be noted that the internal use of function `boot()` implies that the user can easily take advantage of parallel computing to reduce computation time.

Package **robmed** provides various accessor functions to extract relevant information from the returned objects, such as `coef()` and `confint()` methods. In addition, it contains the plot functions `weight_plot()` and `ellipse_plot()` for diagnostics, as well as `ci_plot()` and `density_plot()` for visual representation of the results. All of this functionality is showcased via case studies in the next section.

4. Illustrations: Using package robmed

We demonstrate the use of package **robmed** in three illustrative mediation analyses using the included example data set. We start by loading the package and the data.

```
R> library("robmed")
R> data("BSG2014")
```

The data come from a business strategy game that was played by senior business administration students as part of a course at a Western European university. For more information on the data (including measurement scales), we refer to its R help file, which can be accessed from the console with `?BSG2014`.

We also store the seed to be used for the random number generator in an object, as it will be needed in all examples for the purpose of replicating the results.

```
R> seed <- 20211117
```

The following subsections provide examples for a simple mediation model (Section 4.1), a serial multiple mediator model (Section 4.2), as well as a parallel multiple mediator model with additional control variables (Section 4.3).

4.1. Simple mediation

In the first code example, we replicate parts of the empirical example of Alfons *et al.* (2021). The illustrative hypothesis to be investigated is that task conflict mediates the relationship between team value diversity and team commitment. Using **robmed**'s formula interface (see Section 3.1), we specify a simple mediation model with dependent variable `TeamCommitment` on the left hand side. On the right hand side, we have the hypothesized mediator `TaskConflict`, which is wrapped in a call to function `m()`, as well as the independent variable `ValueDiversity`. As we will compare the robust bootstrap test of Alfons *et al.* (2021) with the OLS bootstrap test (e.g., Preacher and Hayes 2004, 2008; Hayes 2018), we store the formula object for later use.

```
R> f_simple <- TeamCommitment ~ m(TaskConflict) + ValueDiversity
```

Next, we perform the two bootstrap tests using function `test_mediation()`. As usual for functions that fit models, we supply the model specification and the data via the `formula` and `data` arguments. By default, `test_mediation()` fits the mediation model via regressions (argument `method = "regression"`) and performs a bootstrap test for the indirect effect (argument `test = "boot"`). In that case, setting `robust = TRUE` (the default) or `robust = "MM"` specifies the robust bootstrap procedure of Alfons *et al.* (2021), while `robust = FALSE` yields the nonrobust OLS bootstrap test. Before each call to `test_mediation()`, we set the seed of the random number generator.

```
R> set.seed(seed)
R> robust_boot_simple <- test_mediation(f_simple, data = BSG2014,
+                                       robust = TRUE)
R> set.seed(seed)
R> ols_boot_simple <- test_mediation(f_simple, data = BSG2014,
+                                    robust = FALSE)
```

Other estimation methods for a bootstrap test can be specified via a combination of arguments, as outlined in Table 1.

Function `test_mediation()` returns an object inheriting from class `"test_mediation"`. The corresponding `summary()` method shows the relevant information on the fitted models and emphasizes the significance tests of the total, direct, and indirect effects. For bootstrap tests, the information displayed by `summary()` by default stays within the bootstrap framework. For effects other than the indirect effect, asymptotic tests are performed using the normal approximation of the bootstrap distribution. That is, the sample mean and the sample standard deviation of the bootstrap replicates are used for asymptotic z tests. Furthermore, bootstrap estimates of all effects are shown in addition to the estimates on the original data. At the bottom of the output, the indirect effect is summarized by the estimate on the original data (column `Data`), the bootstrap estimate (i.e., the sample mean of the bootstrap replicates; column `Boot`), and the lower and upper limits of the confidence interval (columns `Lower` and `Upper`, respectively).

```
R> summary(robust_boot_simple)

Robust bootstrap test for indirect effect via regression
```

```
x = ValueDiversity
y = TeamCommitment
m = TaskConflict

Sample size: 89
---
Outcome variable: TaskConflict

Coefficients:
                 Data   Boot Std. Error z value Pr(>|z|)
(Intercept)    1.1182 1.1174     0.1798   6.214 5.16e-10 ***
ValueDiversity 0.3197 0.3208     0.1071   2.996  0.00274 **

Robust residual standard error: 0.3033 on 87 degrees of freedom
Robust R-squared:  0.1181,       Adjusted robust R-squared:  0.108
Robust F-statistic: 9.113 on 1 and Inf DF,  p-value: 0.002539

Robustness weights:
4 observations are potential outliers with weight <= 1.3e-05:
[1] 48 58 76 79
---
Outcome variable: TeamCommitment

Coefficients:
                   Data     Boot Std. Error z value Pr(>|z|)
(Intercept)     4.33385  4.33515    0.34415  12.597   <2e-16 ***
TaskConflict   -0.33659 -0.33672    0.17759  -1.896    0.058 .
ValueDiversity  0.06523  0.06388    0.18593   0.344    0.731

Robust residual standard error: 0.3899 on 86 degrees of freedom
Robust R-squared:  0.08994,        Adjusted robust R-squared:  0.06878
Robust F-statistic: 1.497 on 2 and Inf DF,  p-value: 0.2239

Robustness weights:
Observation 6 is a potential outlier with weight 0
---
Total effect of x on y:
                   Data     Boot Std. Error z value Pr(>|z|)
ValueDiversity -0.04239 -0.04293    0.18704   -0.23    0.818

Direct effect of x on y:
                  Data    Boot Std. Error z value Pr(>|z|)
ValueDiversity 0.06523 0.06388    0.18593   0.344    0.731

Indirect effect of x on y:
                Data    Boot   Lower     Upper
TaskConflict -0.1076 -0.1068 -0.2936 -0.009158
```

```
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The above results are very similar to those reported in Alfons *et al.* (2021), but they are not identical due to a recent change in the random number generator in R and the use of a specific seed. Specifically, the robust bootstrap test detects a significant indirect effect, as the confidence interval is strictly negative.

Note that the output corresponding to the regression models is similar to that of the `summary()` method for `"lmrob"` objects from package **robustbase** (Maechler, Rousseeuw, Croux, Todorov, Ruckstuhl, Salibián-Barrera, Verbeke, Koller, Conceição, and Di Palma 2021), but it is shortened as the output is already rather long. In particular, we emphasize that the indices of potential outliers are displayed for each regression model. Those potential outliers should always be investigated further, as they may be interesting observations that could lead to new insights when studied separately (see Alfons *et al.* 2021, for a detailed discussion on outliers in the mediation model).

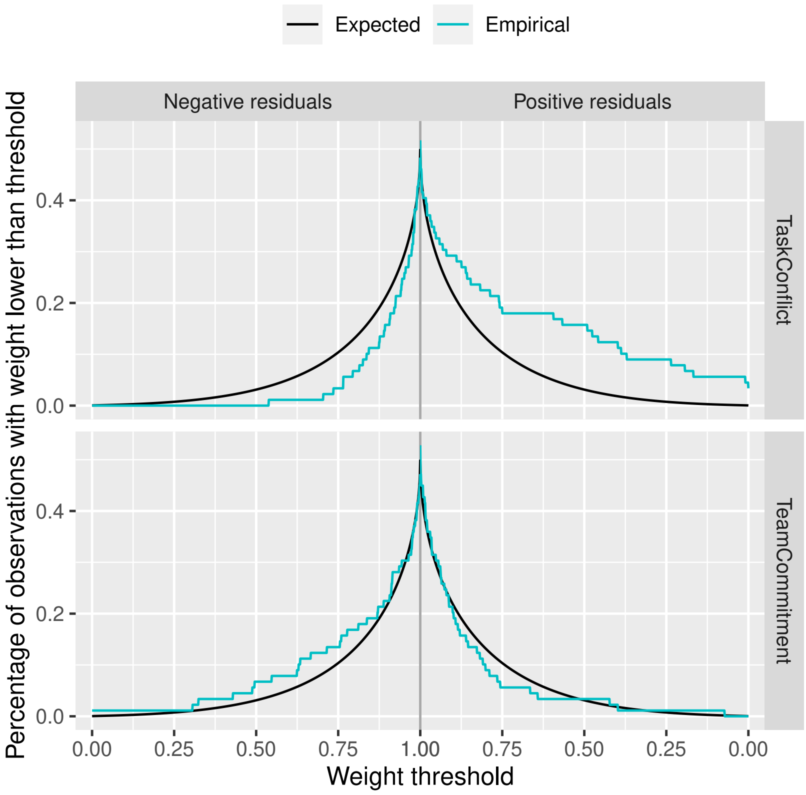

Figure 6: Diagnostic plot of the regression weights from the robust bootstrap procedure of Alfons *et al.* (2021).

Moreover, the summary() method for the robust bootstrap procedure of Alfons *et al.* (2021) by default produces a diagnostic plot that allows to detect deviations from normality assumptions. Keep in mind that this procedure uses the robust MM-estimator of regression (Yohai 1987; Salibián-Barrera and Zamar 2002), which assigns robustness weights to all observations. Those weights can take any value in the interval $[0,1]$, with lower values indicating a higher degree of deviation. For a varying threshold on the horizontal axis, the diagnostic plot displays how many observations have a weight below this threshold. The plot is thereby split into separate panels for negative and positive residuals. For comparison, a reference line is drawn for the expected percentages under normally distributed errors.

Figure 6 shows the plot for this example. For the regression of the hypothesized mediator (TaskConflict) on the independent variable in the top row of the plot, it reveals much more downweighted observations with positive residuals than expected and fewer with negative residuals. This indicates right skewness with a heavy upper tail.

It is possible to suppress the plot when calling summary() by setting plot = FALSE. Function weight_plot() can then be used to create the diagnostic plot. In this example, Figure 6 can also be produced with the commands below.

```
R> weight_plot(robust_boot_simple) +
+    scale_color_manual("", values = c("black", "#00BFC4")) +
+    theme(legend.position = "top")
```

It should also be noted that the output from summary() is structured in a similar way as the output of the widely-used macro PROCESS (Hayes 2018), which implements the OLS bootstrap test for conditional process models such as the mediation model. The intention is to facilitate the use of package **robmed** for users of the PROCESS macro. While PROCESS constructs a bootstrap confidence interval for the indirect effect, it reports estimates on the original data and the usual normal-theory t tests for all other effects. In **robmed**, the same can be achieved by setting the argument type = "data" in the summary() method. The results from the regressions are then summarized in the usual way, as shown below for the OLS bootstrap.

```
R> summary(ols_boot_simple, type = "data")

Bootstrap test for indirect effect via regression

x = ValueDiversity
y = TeamCommitment
m = TaskConflict

Sample size: 89
---
Outcome variable: TaskConflict

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.5007     0.2069   7.253 1.59e-10 ***
ValueDiversity   0.1552     0.1209   1.283    0.203
```

```
Residual standard error: 0.3908 on 87 degrees of freedom
Multiple R-squared:  0.01857,       Adjusted R-squared:  0.007289
F-statistic: 1.646 on 1 and 87 DF,  p-value: 0.2029
---
Outcome variable: TeamCommitment

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     4.49846    0.28806  15.616  < 2e-16 ***
TaskConflict   -0.36412    0.11783  -3.090  0.00269 **
ValueDiversity -0.02088    0.13418  -0.156  0.87671

Residual standard error: 0.4296 on 86 degrees of freedom
Multiple R-squared:  0.1031,       Adjusted R-squared:  0.08227
F-statistic: 4.944 on 2 and 86 DF,  p-value: 0.009279
---
Total effect of x on y:
               Estimate Std. Error t value Pr(>|t|)
ValueDiversity -0.07738    0.13930  -0.555     0.58

Direct effect of x on y:
               Estimate Std. Error t value Pr(>|t|)
ValueDiversity -0.02088    0.13418  -0.156    0.877

Indirect effect of x on y:
                Data     Boot   Lower   Upper
TaskConflict -0.0565 -0.05838 -0.2137 0.02458
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Unlike the robust bootstrap test above, the OLS bootstrap does not detect a significant indirect effect, since the confidence interval covers 0. Due to the influential heavy tail indicated by the diagnostic plot in Figure 6, the results of the robust bootstrap test can be considered more reliable.

Methods for common generic functions to extract information from objects are implemented in **robmed**, such as a `coef()` method to extract the relevant effects of the mediation model, and `confint()` to extract confidence intervals of those effects.

```
R> coef(robust_boot_simple)

          a           b       Total      Direct    Indirect
 0.32077184 -0.33672132 -0.04292728  0.06388476 -0.10681204
```

```
R> confint(robust_boot_simple)

              2.5 %       97.5 %
a         0.1109227  0.530620997
b        -0.6847919  0.011349287
Total    -0.4095137  0.323659113
Direct   -0.3005305  0.428300051
Indirect -0.2936367 -0.009158447
```

For objects corresponding to bootstrap tests (class "boot_test_mediation"), argument type of the coef() method allows to specify whether to extract the bootstrap estimates ("boot", the default) or the estimates on the original data ("data"). Similarly, argument type of the confint() method allows to specify whether the confidence intervals for the effects other than the indirect effect should be bootstrap confidence intervals ("boot", the default) or normal theory intervals based on the original data ("data").

```
R> coef(ols_boot_simple, type = "data")

          a           b       Total      Direct    Indirect
 0.15517748 -0.36412398 -0.07738318 -0.02087934 -0.05650384

R> confint(ols_boot_simple, type = "data")

               2.5 %     97.5 %
a        -0.08521602  0.3955710
b        -0.59836363 -0.1298843
Total    -0.35426561  0.1994992
Direct   -0.28761575  0.2458571
Indirect -0.21371944  0.0245833
```

In addition, argument parm allows to specify which coefficients or confidence intervals to extract.

```
R> coef(robust_boot_simple, parm = "Indirect")

 Indirect
-0.106812

R> confint(robust_boot_simple, parm = "Indirect")

              2.5 %       97.5 %
Indirect -0.2936367 -0.009158447
```

While the bootstrap tests implemented in test_mediation() construct a confidence interval for the indirect effect based on a pre-specified confidence level $1 - \alpha$, function p_value() allows to analyze the bootstrap distribution and extract the smallest significance level α for which the $(1 - \alpha) \cdot 100\%$ confidence interval of the indirect effect does not contain 0.

```
R> p_value(robust_boot_simple, parm = "Indirect")

Indirect
  0.0281

R> p_value(ols_boot_simple, parm = "Indirect")

Indirect
  0.1483
```

Here, the *p* value for the robust bootstrap test shows strong evidence against the null hypothesis of no mediation, whereas the OLS bootstrap fails to do so.

Several plots are implemented to visualize the results of mediation analysis. They can be applied to each object individually, but it is also possible to combine multiple objects from mediation analysis into a list in order to compare different methods graphically. If names are given to the list elements, those names will be used by the plots to identify the different methods. Note that we use the name "ROBMED" here to refer to the robust bootstrap test.

```
R> boot_list <- list("OLS bootstrap" = ols_boot_simple,
+                    "ROBMED" = robust_boot_simple)
```

Function density_plot() plots the density estimates of the indirect effect. It also adds vertical lines for the point estimates and illustrates the confidence intervals by shaded areas. The plot resulting from the following command is displayed in Figure 7.

```
R> density_plot(boot_list)
```

In order to aid with interpretation of the results from mediation analysis, function ci_plot() allows to visualize the point estimates and confidence intervals of selected effects. The direct effect and the indirect effect are plotted by default, but argument parm can be used to specify which effects to plot. Figure 8 contains the plot created by the command below.

```
R> ci_plot(boot_list, parm = c("Total", "Direct", "Indirect"))
```

Finally, function ellipse_plot() produces a bivariate scatterplot together with a tolerance ellipse that illustrates how well the regression results represent the data, exploiting the relationship between regression coefficients and the covariance matrix. For the robust bootstrap procedure of Alfons *et al.* (2021), the robustness weights from the robust regression estimator can be used to compute a weighted sample covariance matrix, from which the tolerance ellipse is computed. It is important to note that such a weighted covariance matrix is not Fisher consistent (that is, the functional form of the estimator applied to the model distribution does not equal the true covariance matrix), as observations are also expected to be downweighted when all observations follow the model. However, it is straightforward to obtain a correction for a Fisher consistent covariance matrix (see Appendix A).

For instance, we can produce such a plot with the independent variable on the horizontal axis and the hypothesized mediator on the vertical axis. In that case the plot represents the results from the regression of the hypothesized mediator on the independent variable. As the independent variable is the only explanatory variable in this regression model, the plot also adds lines representing the respective regression coefficients.

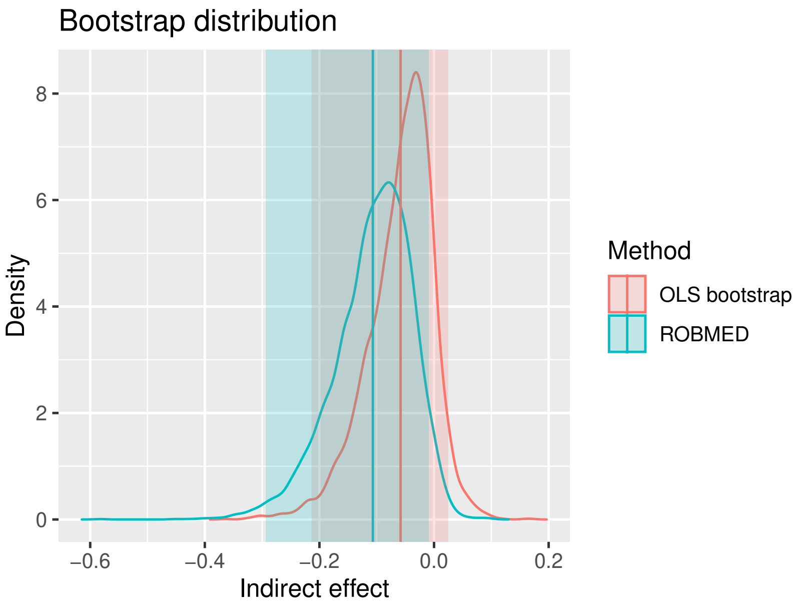

Figure 7: Density plot of the bootstrap distributions of the indirect effect, obtained via the OLS bootstrap and the robust bootstrap procedure of Alfons *et al.* (2021). The vertical lines indicate the the respective point estimates of the indirect effect and the shaded areas represent the confidence intervals.

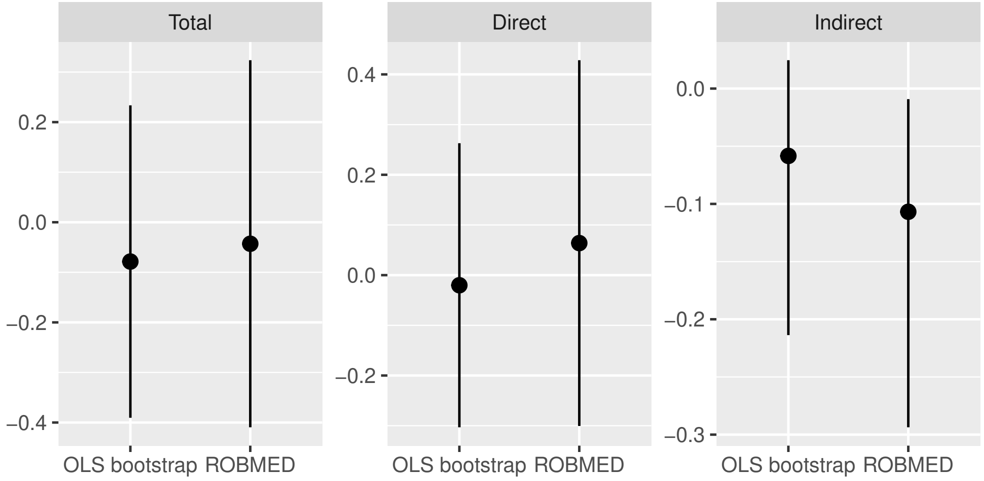

Figure 8: Point estimates and 95% confidence intervals for selected effects in the mediation model, estimated via the OLS bootstrap and the robust bootstrap procedure of Alfons *et al.* (2021).

```
R> ellipse_plot(boot_list, horizontal = "ValueDiversity",
+               vertical = "TaskConflict")
```

The resulting plot is shown in Figure 9. Since we are comparing a nonrobust method with a robust method that assigns a robustness weight to each observation, by default those robust-

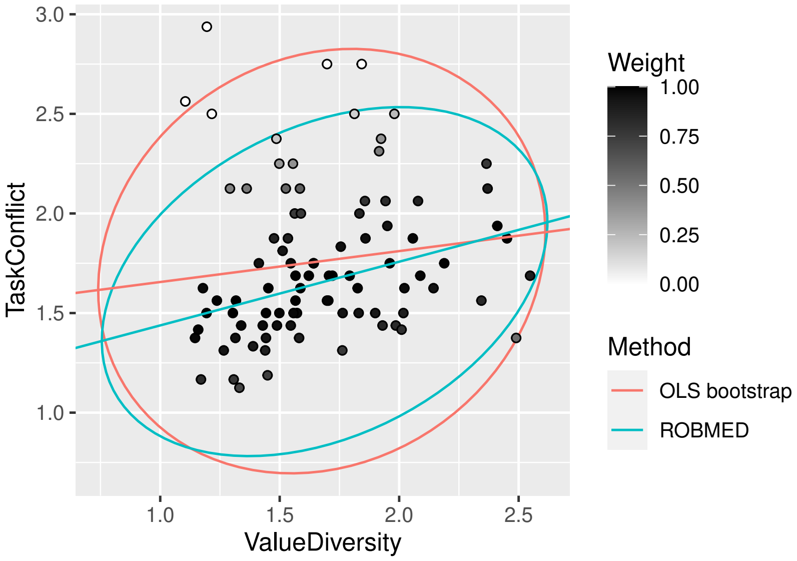

Figure 9: Diagnostic plot with tolerance ellipses for the OLS bootstrap and the robust bootstrap procedure of Alfons *et al.* (2021).

ness weights are visualized by plotting the points on a grayscale. Clearly, the tolerance ellipse corresponding to the robust method fits the main bulk of the data better, as the tolerance ellipse corresponding to the nonrobust method contains more empty space. This plot further suggests that the detected potential outliers (see also the printed output of `summary()` above) are a result of the heavy upper tail in the hypothesized mediator (`TaskConflict`).

All plot functions in **robmed** allow customization via the underlying package **ggplot2** (Wickham 2016). Arguments can be passed down to `geom_xxx()` functions, and additional elements can be added to the plot as usual with the `+` operator. We refer to the R help files of the plots for some examples. Nevertheless, there are limits to the customization, in particular for the plots that contain various elements such as the diagnostic plot with the tolerance ellipse.

For further customization, **robmed** provides the workhorse functions `setup_weight_plot()`, `setup_ci_plot()`, `setup_density_plot()`, and `setup_ellipse_plot()`. They do not produce the plot, but they extract the relevant information to be displayed. They are useful for users who want to create similar plots, but who want more control over what information to display or how to display that information. With the commands below, we manually produce the same plot as before, but only plot the tolerance ellipses and the data without the regression lines. Figure 10 displays the resulting plot.

```
R> setup <- setup_ellipse_plot(boot_list, horizontal = "ValueDiversity",
+                              vertical = "TaskConflict")
R> ggplot() +
+    geom_path(aes(x = x, y = y, color = Method), data = setup$ellipse) +
+    geom_point(aes(x = x, y = y, fill = Weight), data = setup$data,
+               shape = 21) +
+    scale_fill_gradient(limits = 0:1, low = "white", high = "black") +
+    labs(x = setup$horizontal, y = setup$vertical)
```

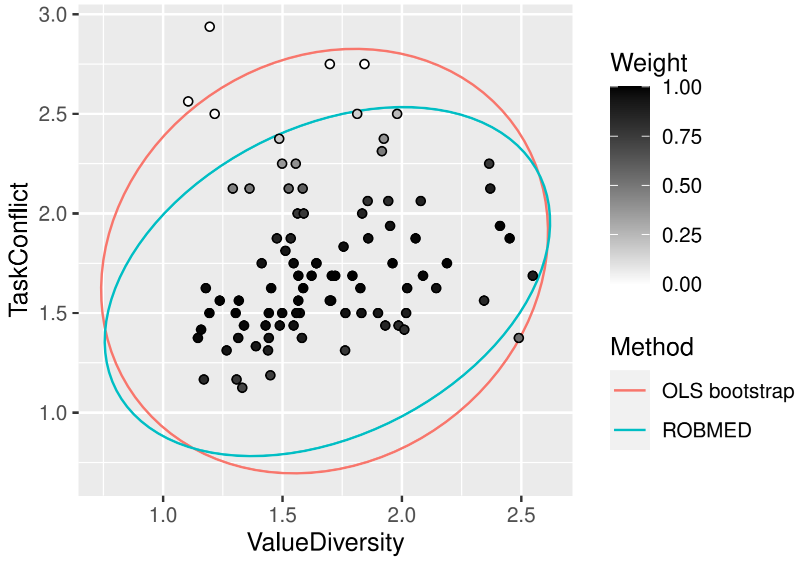

Figure 10: Customized diagnostic plot with tolerance ellipses but without regression lines for the OLS bootstrap and the robust bootstrap procedure of Alfons *et al.* (2021).

4.2. Serial multiple mediators

The second example extends the simple mediation model from the previous section to a serial multiple mediator model. We investigate the following illustrative hypothesis: value diversity negatively impacts team commitment through increased task conflict, and in turn task conflict negatively affects team performance through decreased team commitment. Here the dependent variable is an objective assessment of team performance, measured by the team's score on the simulation game.

The corresponding formula is stored in an object for later use and contains the dependent variable `TeamScore` on the left hand side. On the right hand side, the hypothesized serial mediators `TaskConflict` and `TeamCommitment` are wrapped in a call to `serial_m()`, separated by the usual `+` operator from the independent variable `ValueDiversity`.

```
R> f_serial <- TeamScore ~ serial_m(TaskConflict, TeamCommitment) +
+    ValueDiversity
```

Function `test_mediation()` is then called in the same way as in the previous section to compare the robust bootstrap procedure of Alfons *et al.* (2021) with the nonrobust OLS bootstrap.

```
R> set.seed(seed)
R> robust_boot_serial <- test_mediation(f_serial, data = BSG2014,
+                                       robust = TRUE)
R> set.seed(seed)
R> ols_boot_serial <- test_mediation(f_serial, data = BSG2014,
+                                    robust = FALSE)
```

The output of `summary()` looks very similar to that of the previous example, except that the last part shows results for multiple indirect effects. In order to save space, we only print the objects returned by `test_mediation()`, which shows the results for the bootstrap confidence intervals of the indirect effects. For completeness, the full `summary()` output of the robust bootstrap test can be found in Appendix B.

```
R> robust_boot_serial


Robust bootstrap tests for indirect effects via regression

Indirect effects of x on y:
             Data     Boot  Lower    Upper
Total     -0.2870 -0.46185 -6.859  2.87613
Indirect1  0.1036  0.07264 -1.668  2.24594
Indirect2  0.6010  0.45184 -3.130  5.13150
Indirect3 -0.9916 -0.98633 -3.909 -0.06841

Indirect effect paths:
 Label     Path
 Indirect1 ValueDiversity -> TaskConflict   -> TeamScore
 Indirect2 ValueDiversity -> TeamCommitment -> TeamScore
 Indirect3 ValueDiversity -> TaskConflict   -> TeamCommitment -> TeamScore
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000

R> ols_boot_serial


Bootstrap tests for indirect effects via regression

Indirect effects of x on y:
               Data     Boot  Lower   Upper
Total     -0.427908 -0.52270 -3.530 1.60164
Indirect1  0.009045 -0.11381 -1.179 1.24710
Indirect2 -0.117898 -0.09866 -2.462 1.61605
Indirect3 -0.319055 -0.31024 -1.518 0.08132

Indirect effect paths:
 Label     Path
 Indirect1 ValueDiversity -> TaskConflict   -> TeamScore
 Indirect2 ValueDiversity -> TeamCommitment -> TeamScore
 Indirect3 ValueDiversity -> TaskConflict   -> TeamCommitment -> TeamScore
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
```

Note that the row labeled `Total` in the above output contains the results for the sum of the three individual indirect effects. We observe that the robust bootstrap test detects mediation in the indirect path that goes first through task conflict and subsequently through team commitment (effect `Indirect3`), whereas none of the indirect effects are significant in the OLS bootstrap.

We can use function `weight_plot()` to produce a diagnostic plot to investigate deviations from normality assumptions that could explain the differences in results for the two methods. The plot created with the commands below is shown in Figure 11.

```
R> weight_plot(robust_boot_serial) +
+    scale_color_manual("", values = c("black", "#00BFC4")) +
+    theme(legend.position = "top")
```

As in the previous example, we see that there are strong indications of a heavy upper tail in the regression of task conflict on value diversity (top row of Figure 11) and a heavy lower tail

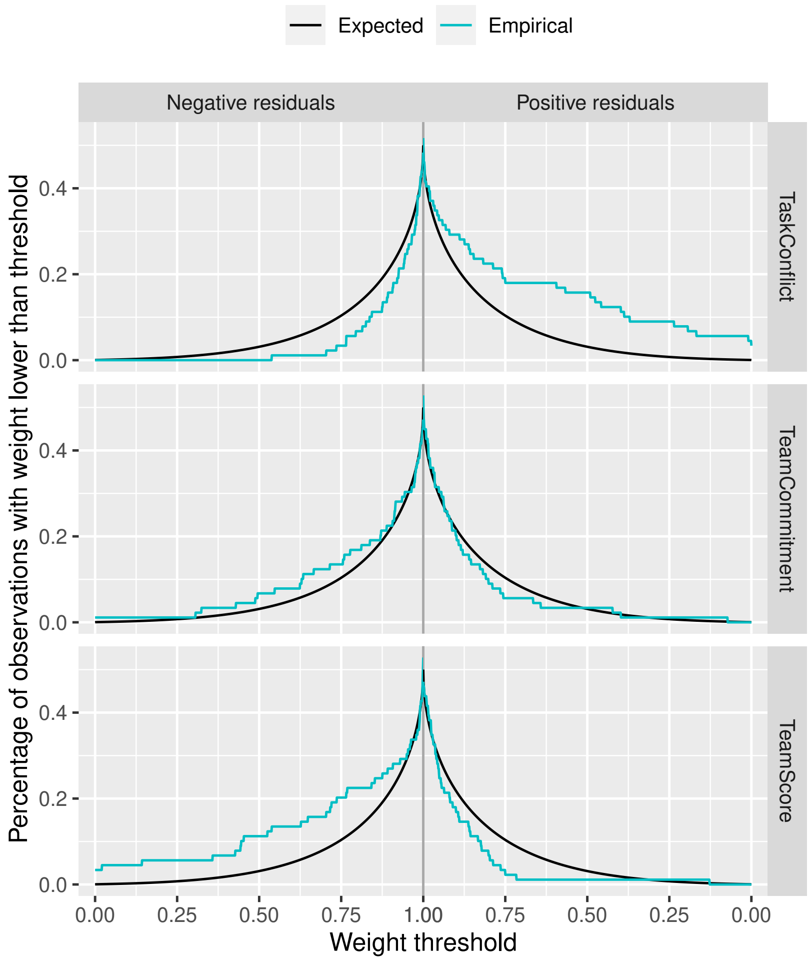

Figure 11: Diagnostic plot of the regression weights from the robust bootstrap procedure of Alfons *et al.* (2021) in the example for a serial multiple mediator model.

in the regression for objective team performance (bottom row of Figure 11), which makes the results of the OLS bootstrap unreliable.

4.3. Parallel multiple mediators and control variables

In the final example, we investigate the illustrative hypothesis that procedural justice and interactional justice are parallel mediators for the relationship between shared leadership and the team's perception of its performance, controlled for diversity in age and gender within the team. Unlike in the previous section, the dependent variable is a subjective measure of team performance, as evaluated by the team members themselves.

In **robmed**'s formula interface, parallel multiple mediators can be specified by wrapping multiple variables in a call to function `parallel_m()`, and control variables can be specified in a similar manner with function `covariates()`.

```
R> f_parallel <-
+    TeamPerformance ~ parallel_m(ProceduralJustice, InteractionalJustice) +
+    SharedLeadership + covariates(AgeDiversity, GenderDiversity)
```

Function `test_mediation()` is then called in the usual way to compare the robust bootstrap test of Alfons *et al.* (2021) and the nonrobust OLS bootstrap.

```
R> set.seed(seed)
R> robust_boot_parallel <- test_mediation(f_parallel, data = BSG2014,
+                                         robust = TRUE)
R> set.seed(seed)
R> ols_boot_parallel <- test_mediation(f_parallel, data = BSG2014,
+                                      robust = FALSE)
```

To save space, we again do not show the entire `summary()` of the resulting objects, but only the results for the indirect effect by printing the objects. Appendix B contains the full `summary()` output of the robust boostrap test, including the diagnostic plot of the regression weights.

```
R> robust_boot_parallel

Robust bootstrap tests for indirect effects via regression

Indirect effects of x on y:
                        Data    Boot     Lower  Upper
Total                0.12089 0.12546 4.398e-02 0.2164
ProceduralJustice    0.03984 0.04568 6.206e-05 0.1184
InteractionalJustice 0.08105 0.07978 3.022e-02 0.1538
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000

R> ols_boot_parallel
```

```
Bootstrap tests for indirect effects via regression

Indirect effects of x on y:
                        Data    Boot    Lower   Upper
Total                0.07452 0.07696 0.015240 0.14838
ProceduralJustice    0.04093 0.04631 0.004064 0.10983
InteractionalJustice 0.03359 0.03064 0.004396 0.08471
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
```

As for the serial multiple mediator model, the row `Total` in the output above contains the results for the sum of the two individual indirect effects through the hypothesized mediators `ProceduralJustice` and `InteractionalJustice`. We observe that both individual indirect effects are significant at the 5% level for both the robust procedure of Alfons *et al.* (2021) and the OLS bootstrap. However, the lower confidence bound for the indirect effect of procedural justice in the robust bootstrap is so small that different seeds of the random number generator often yield a (similarly small) negative lower bound. Overall, we may conclude that we find at least weak support for mediation via procedural justice, and we do find support for mediation via interactional justice.

The output of `summary()` in Appendix B, indicates that the residual distributions are fairly normal (see the diagnostic plot in Figure 13), but that there is a small number of potential outliers that should be investigated further. As the regression of the dependent variable (`TeamPerformance`) on the remaining variables shows more deviations from normality than the other regressions (bottom row of Figure 13), we use `ellipse_plot()` to create the diagnostic plot with a tolerance ellipse that is related to the regression results. If several explanatory variables are included, it is often more insightful to plot the partial residuals on the vertical axis, i.e., to subtract from the response the linear predictor without the variable that is displayed on the horizontal axis. This has the additional advantage that the regression coefficient can be visualized by a line, which is otherwise not possible in the case of multiple explanatory variables. With function `ellipse_plot()`, plotting the partial residuals can easily be achieved by setting the argument `partial = TRUE`. With the command below, we plot the partial residuals of team performance against the independent variable shared leadership. Figure 12 contains the resulting plot, which clearly visualizes the noisy data points.

```
R> ellipse_plot(robust_boot_parallel, horizontal = "SharedLeadership",
+               vertical = "TeamPerformance", partial = TRUE)
```

In multiple mediator models, it can be of interest if the indirect effects are different from one another, or if they differ in magnitude (Hayes 2018, p.163–166). Function `test_mediation()` allows to make pairwise comparisons of indirect effects via the argument `contrast`. By setting this argument to `"estimates"`, the pairwise differences of the estimates of the indirect effect are computed, whereas setting this argument to `"absolute"` yields the computation of pairwise differences in absolute values. The output of `test_mediation()` then includes the estimates and confidence intervals for those contrasts, and also displays information on the definition of the contrasts.

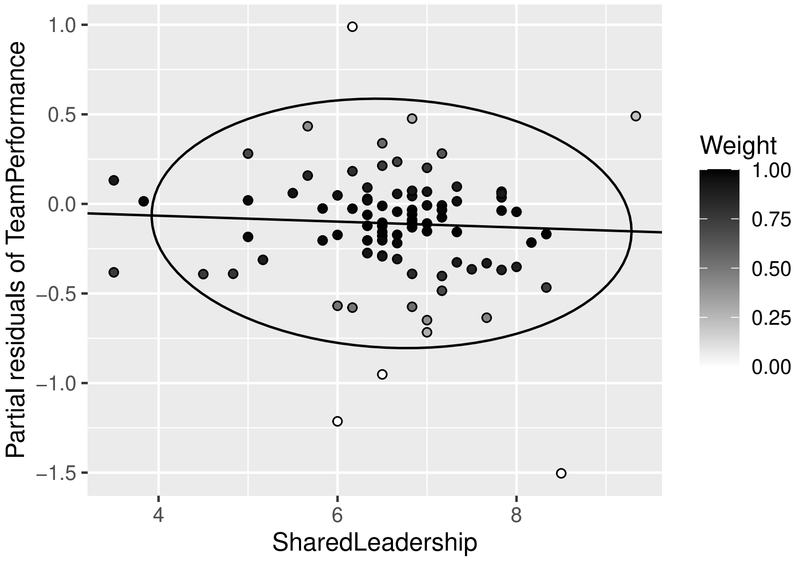

Figure 12: Diagnostic plot with a tolerance ellipse for partial residuals in a multiple parallel mediator model.

```
R> set.seed(seed)
R> test_mediation(f_parallel, data = BSG2014, contrast = "absolute")
```

```
Robust bootstrap tests for indirect effects via regression

Indirect effects of x on y:
                         Data     Boot      Lower   Upper
Total                 0.12089  0.12546  4.398e-02 0.21640
ProceduralJustice     0.03984  0.04568  6.206e-05 0.11841
InteractionalJustice  0.08105  0.07978  3.022e-02 0.15384
Contrast             -0.04121 -0.03393 -1.164e-01 0.05817

Indirect effect contrast definition:
 Label    Definition
 Contrast |ProceduralJustice| - |InteractionalJustice|
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
```

However, it is not actually necessary to run the entire bootstrap procedure again if a bootstrap test had already been performed without computing those contrasts. Function `retest()` allows to reanalyze the bootstrap estimates with different parameter settings, which saves computation time in such a case.

```
R> retest(robust_boot_parallel, contrast = "absolute")
```

```
Robust bootstrap tests for indirect effects via regression

Indirect effects of x on y:
                         Data     Boot      Lower   Upper
Total                 0.12089  0.12546  4.398e-02 0.21640
ProceduralJustice     0.03984  0.04568  6.206e-05 0.11841
InteractionalJustice  0.08105  0.07978  3.022e-02 0.15384
Contrast             -0.04121 -0.03393 -1.164e-01 0.05817

Indirect effect contrast definition:
 Label    Definition
 Contrast |ProceduralJustice| - |InteractionalJustice|
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
```

We emphasize that function `retest()` is available for computational convenience in case the analysis was by mistake conducted with the wrong parameter settings. It must not be abused for p hacking.

5. Summary and discussion

The R package **robmed** provides easy-to-use functionality for robust mediation analysis. It implements the robust bootstrap procedure of Alfons *et al.* (2021), which yields reliable results under outliers and other deviations from normality assumptions, as well as diagnostic plots that allow to detect and futher investigate such deviations. In addition, package **robmed** provides functionality for various other procedures for mediation analysis, as well as plots that allow to visualize and compare results from different methods. All implemented methods thereby share the same function interface and a clear class structure of the results. In particular, **robmed** introduces a new formula interface that allows to specify various types of mediation models with a single formula.

At present, there are some limitations of package **robmed**. First of all, the current version requires a numeric dependent variable and numeric mediators, although the independent variable and additional control variables may be binary or categorical. We aim to extend the fast-and-robust bootstrap methodology for robust estimators of logistic regression in order to add support for mediation analysis with binary dependent variables or mediators. Second, adding support for additional mediation models, such as moderated mediation and mediated moderation models (e.g., Muller *et al.* 2005) is planned for future versions. Finally, a graphical user interface (GUI) could be beneficial for less proficient R users. Developing such a GUI, for instance as a web application based on package **shiny** (Chang, Cheng, Allaire, Sievert, Schloerke, Xie, Allen, McPherson, Dipert, and Borges 2021), is future work. In the meantime, we provide the R extension bundle **ROBMED** (Alfons 2022a) for SPSS (IBM Corp. 2021), which links to the R package **robmed** and allows to use its main functionality through a GUI from within SPSS. Interested readers can obtain the extension bundle from `https://github.com/aalfons/ROBMED-RSPSS`.

Computational details

The results in this paper were obtained using R version 4.1.3 (R Core Team 2022) with package **robmed** version 0.10.1 (Alfons 2022b) and its dependencies **boot** version 1.3.28 (Canty and Ripley 2021), **ggplot2** version 3.3.5 (Wickham 2016), and **robustbase** version 0.93.9 (Maechler *et al.* 2021). Package **robmed** also builds upon packages **quantreg** (Koenker 2022) and **sn** (Azzalini 2022) for functionality not shown in this paper, as well as package **testthat** (Wickham 2011) for unit tests. R itself and all mentioned packages are available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/`. The latest development version of package **robmed** can be obtained from `https://github.com/aalfons/robmed`.

Acknowledgments

Andreas Alfons is supported by a grant of the Dutch Research Council (NWO), research program Vidi, project number VI.Vidi.195.141.

References

Alfons A (2022a). ***ROBMED**: SPSS R Extension Bundle for Robust Mediation Analysis.* `doi:10.25397/eur.19354268.v1`. URL `https://github.com/aalfons/ROBMED-RSPSS`.

Alfons A (2022b). ***robmed**: (Robust) Mediation Analysis.* R package version 0.10.1, URL `https://CRAN.R-project.org/package=robmed`.

Alfons A, Ateş NY, Groenen PJF (2021). "A Robust Bootstrap Test for Mediation Analysis." *Organizational Research Methods*. `doi:10.1177/1094428121999096`.

Asparouhov T, Muthén B (2016). "Structural Equation Models and Mixture Models with Continuous Nonnormal Skewed Distributions." *Structural Equation Modeling: A Multidisciplinary Journal*, **23**(1), 1–19. `doi:10.1080/10705511.2014.947375`.

Azzalini A (2022). ***sn**: The Skew-Normal and Related Distributions Such as the Skew-t and the SUN*. R package version 2.0.2, URL `https://CRAN.R-project.org/package=sn`.

Azzalini A, Arellano-Valle RB (2013). "Maximum Penalized Likelihood Estimation for Skew-Normal and Skew-*t* Distributions." *Journal of Statistical Planning and Inference*, **143**(2), 419–433. `doi:10.1016/j.jspi.2012.06.022`.

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. `doi:10.1037/0022-3514.51.6.1173`.

Bollen KA, Stine R (1990). "Direct and Indirect Effects: Classical and Bootstrap Estimates of Variability." *Sociological Methodology*, **20**, 115–140. `doi:10.2307/271084`.

Canty A, Ripley B (2021). ***boot**: Bootstrap Functions (Originally by Angelo Canty for S)*. R package version 1.3-28, URL `https://CRAN.R-project.org/package=boot`.

Chang W, Cheng J, Allaire JJ, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B (2021). ***shiny**: Web Application Framework for R.* R package version 1.7.1, URL https://CRAN.R-project.org/package=shiny.

Davison AC, Hinkley DV (1997). *Bootstrap Methods and Their Application.* Cambridge University Press, Cambridge, UK. ISBN 0-521-57471-4.

Erreygers S, Vandebosch H, Vranjes I, Baillien E, De Witte H (2018). "The Interplay of Negative Experiences, Emotions and Affective Styles in Adolescents' Cybervictimization: A Moderated Mediation Analysis." *Computers in Human Behavior*, **81**, 223–234. doi:10.1016/j.chb.2017.12.027.

Fox J, Nie Z, Byrnes J (2022). ***sem**: Structural Equation Models.* R package version 3.1-14, URL https://CRAN.R-project.org/package=sem.

Gaudiano BA, Herbert JD, Hayes SC (2010). "Is It the Symptom or the Relation to It? Investigating Potential Mediators of Change in Acceptance and Commitment Therapy for Psychosis." *Behavior Therapy*, **41**(4), 543–554. doi:10.1016/j.beth.2010.03.001.

Hayes AF (2018). *Introduction to Mediation, Moderation, and Conditional Process Analysis.* 2nd edition. The Guilford Press, New York, NY. ISBN 9781462534654.

IBM Corp (2021). *IBM SPSS Statistics, Version 28.0.* Armonk, NY. URL https://www.ibm.com/products/spss-statistics.

Judd CM, Kenny DA (1981). "Process Analysis: Estimating Mediation in Treatment Evaluations." *Evaluation Review*, **5**(5), 602–619. doi:10.1177/0193841X8100500502.

Kelley K (2022). ***MBESS**: The **MBESS** R Package.* R package version 4.9.0, URL https://CRAN.R-project.org/package=MBESS.

Koenker R (2022). ***quantreg**: Quantile Regression.* R package version 5.88, URL https://CRAN.R-project.org/package=quantreg.

MacKinnon DP, Lockwood CM, Hoffman JM, West SG, Sheets V (2002). "A Comparison of Methods to Test Mediation and Other Intervening Variable Effects." *Psychological Methods*, **7**(1), 83–104. doi:10.1037/1082-989x.7.1.83.

MacKinnon DP, Lockwood CM, Williams J (2004). "Confidence Limits for the Indirect Effect: Distribution of the Product and Resampling Methods." *Multivariate Behavioral Research*, **39**(1), 99–128. doi:10.1207/s15327906mbr3901_4.

MacKinnon DP, Warsi G, Dwyer JH (1995). "A Simulation Study of Mediated Effect Measures." *Multivariate Behavioral Research*, **30**(1), 41–62. doi:10.1207/s15327906mbr3001_3.

Maechler M, Rousseeuw P, Croux C, Todorov V, Ruckstuhl A, Salibián-Barrera M, Verbeke T, Koller M, Conceição ELT, Di Palma MA (2021). ***robustbase**: Basic Robust Statistics.* R package version 0.93-9, URL https://CRAN.R-project.org/package=robustbase.

Mair P, Wilcox R (2020). "Robust Statistical Methods in R Using the **WRS2** Package." *Behavior Research Methods*, **52**(2), 464–488. doi:10.3758/s13428-019-01246-w.

Muller D, Judd CM, Yzerbyt VY (2005). "When Moderation Is Mediated and Mediation Is Moderated." *Journal of Personality and Social Psychology*, **89**(6), 852–864. doi:10.1037/0022-3514.89.6.852.

Muthén LK, Muthén BO (2017). *Mplus User's Guide*. Los Angeles, CA. 8th edition, URL https://www.statmodel.com/.

Preacher KJ, Hayes AF (2004). "SPSS and SAS Procedures for Estimating Indirect Effects in Simple Mediation Models." *Behavior Research Methods, Instruments, & Computers*, **36**(4), 717–731. doi:10.3758/BF03206553.

Preacher KJ, Hayes AF (2008). "Asymptotic and Resampling Strategies for Assessing and Computing Indirect Effects in Multiple Mediator Models." *Behavior Research Methods*, **40**(3), 879–891. doi:10.3758/BRM.40.3.879.

SAS Institute Inc (2020). *SAS/STAT 15.2*. Cary, NC. URL https://www.sas.com/.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Revelle W (2021). ***psych**: Procedures for Psychological, Psychometric, and Personality Research*. R package version 2.1.9, URL https://CRAN.R-project.org/package=psych.

Rosseel Y (2012). "**lavaan**: An R Package for Structural Equation Modeling." *Journal of Statistical Software*, **48**(2), 1–36. doi:10.18637/jss.v048.i02.

Rousseeuw PJ, Yohai VJ (1984). "Robust Regression by Means of S-Estimators." In J Franke, W Härdle, D Martin (eds.), *Robust and Nonlinear Time Series Analysis*, volume 26 of *Lecture Notes in Statistics*, pp. 256–272. Springer-Verlag, New York, NY.

Salibián-Barrera M, Van Aelst S (2008). "Robust Model Selection Using Fast and Robust Bootstrap." *Computational Statistics & Data Analysis*, **52**(12), 5121–5135. doi:10.1016/j.csda.2008.05.007.

Salibián-Barrera M, Yohai VJ (2006). "A Fast Algorithm for S-Regression Estimates." *Journal of Computational and Graphical Statistics*, **15**(2), 414–427. doi:10.1198/106186006X113629.

Salibián-Barrera M, Zamar RH (2002). "Bootstrapping Robust Estimates of Regression." *The Annals of Statistics*, **30**(2), 556–582. doi:10.1214/aos/1021379865.

Schwarz G (1978). "Estimating the Dimension of a Model." *The Annals of Statistics*, **6**(2), 461–464. doi:10.1214/aos/1176344136.

Shrout PE, Bolger N (2002). "Mediation in Experimental and Nonexperimental Studies: New Procedures and Recommendations." *Psychological Methods*, **7**(4), 422–445. doi:10.1037/1082-989X.7.4.422.

Sobel ME (1982). "Asymptotic Confidence Intervals for Indirect Effects in Structural Equation Models." *Sociological Methodology*, **13**, 290–312. doi:10.2307/270723.

Steen J, Loeys T, Moerkerke B, Vansteelandt S (2017). "**medflex**: An R Package for Flexible Mediation Analysis Using Natural Effect Models." *Journal of Statistical Software*, **76**(11), 1–46. doi:10.18637/jss.v076.i11.

Tingley D, Yamamoto T, Hirose K, Keele L, Imai K (2014). "**mediation**: R Package for Causal Mediation Analysis." *Journal of Statistical Software*, **59**(5), 1–38. doi:10.18637/jss.v059.i05.

Vuorre M (2021). ***bmlm**: Bayesian Multilevel Mediation*. R package version 1.3.12, URL https://CRAN.R-project.org/package=bmlm.

Wickham H (2011). "**testthat**: Get Started with Testing." *The R Journal*, **3**(1), 5–10. doi:10.32614/RJ-2011-002.

Wickham H (2016). ***ggplot2**: Elegant Graphics for Data Analysis*. Springer-Verlag, New York, NY. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org.

Wood RE, Goodman JS, Beckmann N, Cook A (2008). "Mediation Testing in Management Research: A Review and Proposals." *Organizational Research Methods*, **11**(2), 270–295. doi:10.1177/1094428106297811.

Yohai VJ (1987). "High Breakdown-Point and High Efficiency Robust Estimates for Regression." *The Annals of Statistics*, **15**(2), 642–656. doi:10.1214/aos/1176350366.

Yu Q, Li B (2021). ***mma**: Multiple Mediation Analysis*. R package version 10.5-1, URL https://CRAN.R-project.org/package=mma.

Yuan Y, MacKinnon DP (2014). "Robust Mediation Analysis Based on Median Regression." *Psychological Methods*, **19**(1), 1–20. doi:10.1037/a0033820.

Zu J, Yuan KH (2010). "Local Influence and Robust Procedures for Mediation Analysis." *Multivariate Behavioral Research*, **45**(1), 1–44. doi:10.1080/00273170903504695.

A. Details on the diagnostic plot with a tolerance ellipse

The diagnostic plot from function `ellipse_plot()` exploits the relationship between regression coefficients and the covariance matrix to draw a tolerance ellipse that illustrates how well the regression results represent the data. Our recommended robust bootstrap test for mediation analysis is based on a robust regression estimator that assigns robustness weights to the observations. Those robustness weights lie between 0 and 1, with lower weights indicating a higher degree of deviation. The corresponding tolerance ellipse is computed based on a weighted sample covariance matrix, using the weights returned by the robust regression. However, for such a plot to be meaningful, the weighted sample covariance matrix needs to yield a Fisher consistent estimator of the true covariance matrix under the model distribution.

The diagnostic plot is most useful if the data come from a multivariate elliptical distribution. Consider the regression model

$$Y = \alpha + \boldsymbol{X}^\top \boldsymbol{\beta} + \sigma\varepsilon,$$

In addition to the usual assumptions that $\varepsilon \sim N(0, 1)$ and that $\boldsymbol{X}$ and ε are uncorrelated, we therefore also assume for the diagnostic plot that $\boldsymbol{X} \sim N(\boldsymbol{\mu}_{\boldsymbol{X}}, \boldsymbol{\Sigma}_{\boldsymbol{XX}})$. We emphasize that this additional assumption is made only for the diagnostic plot, it is not required for the general applicability of our robust bootstrap test. Then we have $Z = (Y, \boldsymbol{X}^\top)^\top \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_Y \\ \boldsymbol{\mu}_{\boldsymbol{X}} \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \Sigma_{YY} & \boldsymbol{\Sigma}_{Y\boldsymbol{X}} \\ \boldsymbol{\Sigma}_{\boldsymbol{X}Y} & \boldsymbol{\Sigma}_{\boldsymbol{XX}} \end{pmatrix}.$$

Furthermore, the regression coefficients can be written as

$$\beta = \boldsymbol{\Sigma}_{\boldsymbol{XX}}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{X}Y},$$
$$\alpha = \mu_Y - \boldsymbol{\mu}_{\boldsymbol{X}}^\top \boldsymbol{\beta}.$$

With observations $\mathbf{z}_i = (y_i, \mathbf{x}_i^\top)^\top$ and with with robustness weights w_i from robust regression, $i = 1, \ldots, n$, we define the weighted center estimate $\mathbf{m}$ and the weighted scatter matrix $\boldsymbol{S}$ as

$$\mathbf{m} = \frac{1}{\sum_{i=1}^n w_i} \sum_{i=1}^n w_i \mathbf{z}_i,$$
$$\boldsymbol{S} = \frac{1}{(\sum_{i=1}^n w_i) - 1} \sum_{i=1}^n w_i (\mathbf{z}_i - \mathbf{m})(\mathbf{z}_i - \mathbf{m})^\top.$$

Note that the denominator of $\boldsymbol{S}$ is chosen such that the weighted scatter matrix reduces to the unbiased sample covariance matrix if all observations receive full weight $w_i = 1$, $i = 1, \ldots, n$. Let F denote the cumulative distribution function (CDF) of the multivariate normal distribution $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and let Φ denote the CDF of the univariate standard normal distribution $N(0, 1)$. Then the functional forms of the estimators are given by

$$\boldsymbol{m}(F) = \frac{1}{\delta} \int w(\varepsilon) \boldsymbol{z} \, dF(\boldsymbol{z}),$$
$$\boldsymbol{S}(F) = \frac{1}{\delta} \int w(\varepsilon)(\boldsymbol{z} - \boldsymbol{m}(F))(\boldsymbol{z} - \boldsymbol{m}(F))^\top \, dF(\boldsymbol{z}),$$

where $\delta = \int w(\varepsilon) \, d\Phi(\varepsilon)$.

Keep in mind that we only consider robust regression with symmetric loss functions such that the weight function w is also symmetric, and let $\boldsymbol{m}(F) = (m_Y(F), \boldsymbol{m}_{\boldsymbol{X}}(F)^\top)^\top$. For the explanatory variables $\boldsymbol{X}$, we have

$$\boldsymbol{m}_{\boldsymbol{X}}(F) = \frac{1}{\delta} \int w(\varepsilon) \boldsymbol{x} \, dF(\boldsymbol{z}) = \frac{1}{\delta} \underbrace{\int w(\varepsilon) \, d\Phi(\varepsilon)}_{=\delta} \int \boldsymbol{x} \, dF_{\boldsymbol{X}}(\boldsymbol{x}) = \boldsymbol{\mu}_{\boldsymbol{X}},$$

where $F_{\boldsymbol{X}}$ denotes the joint CDF of $\boldsymbol{X}$. To show that $m_Y(F) = \mu_Y$, we can assume without loss of generality that $\mu = (0, \dots, 0)^\top$ and that $\boldsymbol{\beta} = (0, \dots, 0)^\top$. Then $\alpha = 0$ and $\varepsilon = Y/\sigma$, and we need to show that $m_Y(F) = 0$. We obtain

$$m_Y(F) = \frac{1}{\delta} \int w(\varepsilon) y \, dF(\boldsymbol{z}) = \frac{1}{\delta} \int_{-\infty}^{\infty} \underbrace{w\left(\frac{y}{\sigma}\right) y f_Y(y)}_{=g(y)} \, dy, \tag{19}$$

where f_Y denotes the probability density function of the marginal distribution of Y. For the integrand in (19), it holds that $g(-y) = -g(y)$, hence we have $m_Y(F) = 0$. Thus $\hat{\boldsymbol{\mu}} = \boldsymbol{m}$ is Fisher consistent for $\boldsymbol{\mu}$.

Since $0 \leq w(\varepsilon) \leq 1$, $\boldsymbol{S}(F)$ is expected to underestimate (some elements of) the covariance matrix $\boldsymbol{\Sigma}$. However, for the submatrix involving only the explanatory variables $\boldsymbol{X}$, we obtain

$$\begin{aligned} \boldsymbol{S}_{\boldsymbol{X}\boldsymbol{X}}(F) &= \frac{1}{\delta} \int w(\varepsilon)(\boldsymbol{x} - \underbrace{\boldsymbol{m}_{\boldsymbol{X}}(F)}_{=\boldsymbol{\mu}_{\boldsymbol{X}}})(\boldsymbol{x} - \underbrace{\boldsymbol{m}_{\boldsymbol{X}}(F)}_{=\boldsymbol{\mu}_{\boldsymbol{X}}})^\top \, dF(\boldsymbol{z}) \\ &= \frac{1}{\delta} \underbrace{\int w(\varepsilon) \, d\Phi(\varepsilon)}_{=\delta} \int (\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{X}})(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{X}})^\top \, dF_{\boldsymbol{X}}(\boldsymbol{x}) = \boldsymbol{\Sigma}_{\boldsymbol{X}\boldsymbol{X}}. \end{aligned}$$

Thus

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}\boldsymbol{X}} = \boldsymbol{S}_{\boldsymbol{X}\boldsymbol{X}} = \frac{1}{(\sum_{i=1}^n w_i) - 1} \sum_{i=1}^n w_i (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$$

is Fisher consistent for $\boldsymbol{\Sigma}_{\boldsymbol{X}\boldsymbol{X}}$. With Fisher consistent estimators $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}$ from robust regression, we can therefore construct a Fisher consistent estimator $\hat{\boldsymbol{\Sigma}}$ of the covariance matrix $\boldsymbol{\Sigma}$ by computing

$$\begin{aligned} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}Y} &= \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}\boldsymbol{X}} \hat{\boldsymbol{\beta}}, \\ \hat{\Sigma}_{YY} &= \hat{\boldsymbol{\beta}}^\top \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}\boldsymbol{X}} \hat{\boldsymbol{\beta}} + \hat{\sigma}^2, \\ \hat{\boldsymbol{\Sigma}} &= \begin{pmatrix} \hat{\Sigma}_{YY} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}Y}^\top \\ \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}Y} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{X}\boldsymbol{X}} \end{pmatrix}. \end{aligned}$$

B. Additional summary output of robust bootstrap tests

We first produce the summary of the robust bootstrap test for the serial multiple mediator model from Section 4.2. Since the diagnostic plot for this example is already shown in Figure 11, we set the argument `plot = FALSE` to suppress the diagnostic plot.

```
R> summary(robust_boot_serial, plot = FALSE)

Robust bootstrap tests for indirect effects via regression

Serial multiple mediator model

x  = ValueDiversity
y  = TeamScore
m1 = TaskConflict
m2 = TeamCommitment

Sample size: 89
---
Outcome variable: TaskConflict

Coefficients:
                 Data   Boot Std. Error z value Pr(>|z|)
(Intercept)    1.1182 1.1171     0.1794   6.226 4.77e-10 ***
ValueDiversity 0.3197 0.3208     0.1071   2.994  0.00275 **

Robust residual standard error: 0.3033 on 87 degrees of freedom
Robust R-squared:  0.1181,        Adjusted robust R-squared:  0.108
Robust F-statistic: 9.113 on 1 and Inf DF,  p-value: 0.002539

Robustness weights:
4 observations are potential outliers with weight <= 1.3e-05:
[1] 48 58 76 79
---
Outcome variable: TeamCommitment

Coefficients:
                   Data     Boot Std. Error z value Pr(>|z|)
(Intercept)     4.33385  4.33709    0.34436  12.595   <2e-16 ***
TaskConflict   -0.33659 -0.33722    0.17763  -1.898   0.0576 .
ValueDiversity  0.06523  0.06319    0.18722   0.338   0.7357

Robust residual standard error: 0.3899 on 86 degrees of freedom
Robust R-squared:  0.08994,        Adjusted robust R-squared:  0.06878
Robust F-statistic: 1.497 on 2 and Inf DF,  p-value: 0.2239

Robustness weights:
Observation 6 is a potential outlier with weight 0
---
Outcome variable: TeamScore

Coefficients:
                  Data    Boot Std. Error z value Pr(>|z|)
```

```
(Intercept)    61.9660 60.7349    16.4148   3.700 0.000216 ***
TaskConflict    0.3240  0.1877     2.7909   0.067 0.946369
TeamCommitment  9.2138  9.4999     4.5906   2.069 0.038508 *
ValueDiversity  0.2024  0.4193     3.0612   0.137 0.891065

Robust residual standard error: 8.42 on 85 degrees of freedom
Robust R-squared:  0.1746,        Adjusted robust R-squared:  0.1455
Robust F-statistic: 1.194 on 3 and Inf DF,  p-value: 0.3103

Robustness weights:
3 observations are potential outliers with weight <= 0:
[1] 10 32 38
---
Total effect of x on y:
                   Data     Boot Std. Error z value Pr(>|z|)
ValueDiversity -0.08467 -0.04260    3.28764  -0.013     0.99

Direct effect of x on y:
                 Data   Boot Std. Error z value Pr(>|z|)
ValueDiversity 0.2024 0.4193     3.0612   0.137    0.891

Indirect effects of x on y:
             Data     Boot  Lower    Upper
Total     -0.2870 -0.46185 -6.859  2.87613
Indirect1  0.1036  0.07264 -1.668  2.24594
Indirect2  0.6010  0.45184 -3.130  5.13150
Indirect3 -0.9916 -0.98633 -3.909 -0.06841

Indirect effect paths:
 Label     Path
 Indirect1 ValueDiversity -> TaskConflict   -> TeamScore
 Indirect2 ValueDiversity -> TeamCommitment -> TeamScore
 Indirect3 ValueDiversity -> TaskConflict   -> TeamCommitment -> TeamScore
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Finally, we display the summary of the robust bootstrap test for the parallel multiple mediator model with control variables from Section 4.3. Figure 13 contains the resulting diagnostic plot.

```
R> summary(robust_boot_parallel)
```

```
Robust bootstrap tests for indirect effects via regression

Parallel multiple mediator model

x  = SharedLeadership
y  = TeamPerformance
m1 = ProceduralJustice
m2 = InteractionalJustice

Covariates:
[1] AgeDiversity    GenderDiversity

Sample size: 89
---
Outcome variable: ProceduralJustice

Coefficients:
                   Data    Boot Std. Error z value Pr(>|z|)
(Intercept)      3.37132 3.34898    0.22221  15.071   <2e-16 ***
SharedLeadership 0.06287 0.06732    0.03632   1.853   0.0638 .
AgeDiversity     0.05232 0.04914    0.03020   1.627   0.1037
GenderDiversity  0.22030 0.20656    0.14977   1.379   0.1678

Robust residual standard error: 0.2377 on 85 degrees of freedom
Robust R-squared:  0.1277,        Adjusted robust R-squared:  0.09696
Robust F-statistic: 1.208 on 3 and Inf DF,  p-value: 0.3052

Robustness weights:
No potential outliers with weight < 0.0011 detected.
The minimum weight is 0.27.
---
Outcome variable: InteractionalJustice

Coefficients:
                     Data      Boot Std. Error z value Pr(>|z|)
(Intercept)       3.276732  3.261434   0.261999  12.448  < 2e-16 ***
SharedLeadership  0.151376  0.153656   0.034581   4.443 8.85e-06 ***
AgeDiversity     -0.007949 -0.005299   0.050795  -0.104    0.917
GenderDiversity   0.344611  0.338612   0.184200   1.838    0.066 .

Robust residual standard error: 0.2788 on 85 degrees of freedom
Robust R-squared:  0.2847,        Adjusted robust R-squared:  0.2595
Robust F-statistic: 2.169 on 3 and Inf DF,  p-value: 0.08934

Robustness weights:
2 observations are potential outliers with weight <= 0:
[1] 31 57
```

```
---
Outcome variable: TeamPerformance

Coefficients:
                          Data       Boot Std. Error z value Pr(>|z|)
(Intercept)         -0.8007568 -0.8296269  0.6028351  -1.376  0.16876
ProceduralJustice    0.6335972  0.6527230  0.2007240   3.252  0.00115 **
InteractionalJustice  0.5354262  0.5247245  0.1754996   2.990  0.00279 **
SharedLeadership    -0.0164857 -0.0157296  0.0504240  -0.312  0.75508
AgeDiversity         0.0025378  0.0008948  0.0362812   0.025  0.98032
GenderDiversity      0.2756176  0.2716191  0.1549740   1.753  0.07966 .

Robust residual standard error: 0.2558 on 83 degrees of freedom
Robust R-squared:  0.5988,       Adjusted robust R-squared:  0.5746
Robust F-statistic: 3.476 on 5 and Inf DF,  p-value: 0.003833

Robustness weights:
3 observations are potential outliers with weight <= 0:
[1] 33 38 79
---
Total effect of x on y:
                    Data    Boot Std. Error z value Pr(>|z|)
SharedLeadership 0.10440 0.10973    0.05295   2.072   0.0382 *

Direct effect of x on y:
                     Data     Boot Std. Error z value Pr(>|z|)
SharedLeadership -0.01649 -0.01573    0.05042  -0.312    0.755

Indirect effects of x on y:
                        Data    Boot     Lower  Upper
Total                0.12089 0.12546 4.398e-02 0.2164
ProceduralJustice    0.03984 0.04568 6.206e-05 0.1184
InteractionalJustice 0.08105 0.07978 3.022e-02 0.1538
---
Level of confidence: 95 %

Number of bootstrap replicates: 5000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

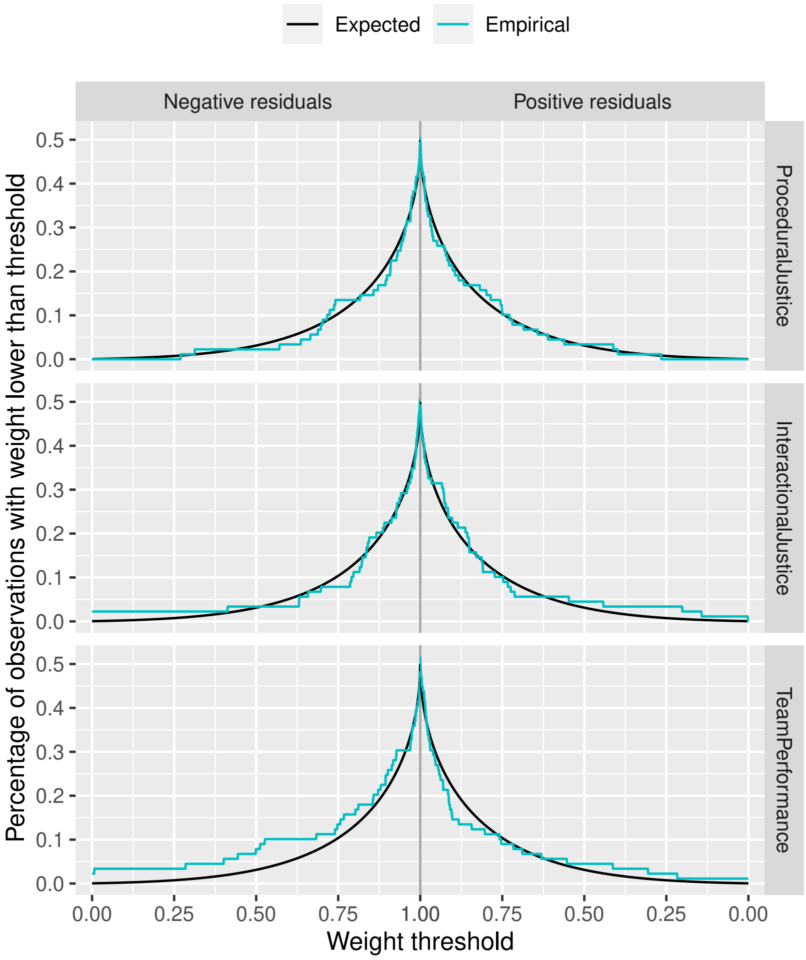

Figure 13: Diagnostic plot of the regression weights from the robust bootstrap procedure of Alfons *et al.* (2021) in the example for a parallel multiple mediator model.

Affiliation:

Andreas Alfons
Econometric Institute
Erasmus School of Economics
Erasmus University Rotterdam
PO Box 1738
3000DR Rotterdam, The Netherlands
E-mail: `alfons@ese.eur.nl`
URL: `https://personal.eur.nl/alfons/`