Two Cross-Platform Programs for Inferences and Interval Estimation About Indirect Effects in Mediational Models

Carl F. Falk and Jeremy C. Biesanz

Abstract
In this article, we describe two new programs that compute both p-values and confidence intervals (CI) for the indirect effect in mediational models, including (a) a p-value based on the partial posterior method, which we refer to as \( p_3 \), computed across the posterior distribution of the regression coefficients; (b) a variant of \( p_3 \) that uses a normal approximation for the posterior distributions, \( p_{3N} \); (c) Hierarchical Bayesian CIs (CI \(_{HB} \)) based on the posterior distributions of the regression coefficients; and (d) CIs based on the Monte Carlo method (CI \(_{MC} \)). These programs do not require access to raw data as do resampling methods. Similar to Sobel’s test, \( p_3 \) and \( p_{3N} \) constitute a single p-value for the indirect effect while performing substantially better in terms of Type I and II error rates. Furthermore, we include a memory efficient computational algorithm for CI \(_{HB} \) and CI \(_{MC} \) that allows for precision beyond that in existing alternative implementations. The underlying programs can utilize multicores processors, and their performance is tested through a simulation study. Finally, the use of these programs is illustrated with an empirical example.

Keywords
mediation analysis, indirect effect, confidence intervals, inference

Researchers in the social sciences are often interested in explaining the relationship between an independent variable and dependent variable in terms of a third process or mediating variable. That is, it is often hypothesized that the independent variable \( (X) \) causes changes in the mediating variable \( (M) \), which in turn causes changes in the dependent variable \( (Y) \). For example, Dunn, Biesanz, Human, and Finn (2007) proposed that the relationship between social affiliation and positive affect can be explained, in part, by self-presentation. Such research questions entail a model such as that depicted in Figure 1. The causal chain, \( X \rightarrow M \rightarrow Y \), is represented by the two coefficients \( a \) and \( b \), and the product, \( ab \), is known as the indirect effect of \( X \) on \( Y \) that passes through \( M \). In this article, we will refer to \( \hat{a} \) and \( \hat{b} \) as the sample estimates of these coefficients.

A variety of statistical procedures exist for making inferences and confidence intervals (CIs) for the indirect effect, including traditional/asymptotic approaches (Baron & Kenny, 1986; Sobel, 1982), bootstrapping/resampling methods (Shrout & Bolger, 2002), an analytical approximation to the distribution of the product (DPR) of \( \hat{a} \) and \( \hat{b} \) (MacKinnon, Fritz, Williams, & Lockwood, 2007), a Monte Carlo approximation to the DPR (Preacher & Selig, 2012), and so on. Each of these approaches has its own advantages/disadvantages in terms of empirical performance (Type I and Type II error rates), computational ease, interpretation, and software availability. The overarching goal of the present research is to provide applied researchers easy access to methods for intervals and inference for indirect effects that have empirical performance that is as good as or better than available alternatives. A secondary goal is to adapt a computational algorithm for CIs that allows greater arbitrary precision of each bound than some alternative implementations of the Monte Carlo method. To achieve these goals, we present two open-source software programs that implement p-values and CIs for the indirect effect.

The p-values are based on the partial posterior method (Biesanz, Falk, & Savalei, 2010), which we refer to as \( p_3 \), and come in two flavors: (a) \( p_3 \) computed using the posterior distribution of the regression coefficients (appropriate for multiple regression models) and (b) \( p_{3N} \) computed using a normal approximation to all posterior and sampling distributions (appropriate for structural equation models, possibly including latent variables; for example, Falk & Biesanz, 2015). \( p_3 \) constitutes a p-value for the indirect effect and thus

1Michigan State University, East Lansing, USA
2University of British Columbia, Vancouver, Canada

Corresponding Author:
Carl F. Falk, Measurement and Quantitative Methods, Michigan State University, 458 Erickson Hall, East Lansing, MI 48824, USA.
Email: falkcarl@msu.edu
should be considered a potential replacement for the popular Sobel’s test. Recent results from a large-scale simulation study suggest that the partial posterior method has high power while maintaining good Type I error control even under nonnormal data (Biesanz et al., 2010).

Similarly, CIs come in two flavors: (a) Hierarchical Bayesian CIs (Biesanz et al., 2010), CI_{HB}, and (b) Monte Carlo CIs, CI_{MC} (e.g., Preacher & Selig, 2012). While the former have been implemented for use with regression models, Monte Carlo CIs may be used when a normal approximation is reasonable for the sampling distribution of the regression coefficients (e.g., as in structural equation models). Whereas CI_{HB} also performs at or near the top in terms of coverage rates (i.e., containing the indirect effect within the CI) and Type I error control, CI_{MC} should be asymptotically equivalent to the DPR method (MacKinnon, Fritz, et al., 2007), which also performs well (Biesanz et al., 2010).1

While our implemented methods perform at or near the top in terms of Type I and Type II errors, they also do not necessarily require access to the raw data (unlike nonparametric bootstrapping or resampling methods) nor refitting of the mediational model (as required by parametric and nonparametric bootstrapping). Information necessary for computing most of the methods our software implements (with the exception of some special cases with the Monte Carlo method) is similar to that required by Sobel’s test. Prior to our work, all but the Monte Carlo method have not been made readily available to applied researchers. The underlying methods are also made available using open-source Java programs. This implementation serves two purposes. First, our use of Java allows for a graphical user-interface that may be appealing to some, and allows researchers to use the programs in conjunction with the statistical software and operating system of their choice. For example, whereas alternative programs may have many options in terms of the complexity of the allowed mediational model (Hayes, 2012), they are tied to the use of specific commercial software (i.e., SPSS and SAS) and cannot handle some unsupported models that work with our implemented methods (e.g., P3N and CI_{MC} for structural equation models with latent variables and certain types of hierarchical linear models). Second, making the source code available allows other researchers to modify the program to suit their needs or to conduct additional innovations. For example, the CIs are implemented using a memory efficient stochastic approximation algorithm (Chen, Lambert, & Pinheiro, 2000; Tierney, 1983) that has potentially greater arbitrary precision than existing implementations (e.g., Hicks & Tingley, 2011; Imai, Keele, Tingley, & Yamamoto, 2010; Selig & Preacher, 2008; Tofghi & MacKinnon, 2011). Because the Monte Carlo method is flexible in situations when the desired indirect effect comprises more than just a simple product of two coefficients (Preacher & Selig, 2012), future iterations of the underlying code will easily be adapted to more complex cases such as multiple mediators, moderated mediation, noncontinuous mediators and outcomes (e.g., Iacobucci, 2012), or to test the performance of this method versus alternatives when conducting sensitivity analysis (e.g., Imai, Keele, & Tingley, 2010; Imai, Keele, Tingley, & Yamamoto, 2010).

The remainder of this manuscript is organized as follows. We first present traditional approaches to mediation analysis (e.g., Baron & Kenny, 1986; Sobel, 1982) and briefly review literature on recent methodological developments. We then describe the methods implemented in our cross-platform programs. This is followed by a simulation study showing that our program is operating correctly with these methods performing well against resampling methods. Finally, we illustrate use of the programs and conclude with future developments.

**Literature Review**

Baron and Kenny’s (1986) approach to mediation analysis is one of the most cited articles in the past 30 years in all of social/personality psychology (Nosek et al., 2010) and is often accompanied by a z test or CI based on Sobel’s standard error (Sobel, 1982). In brief, the Baron and Kenny approach (see Figure 2) involves a series of models: (a) predict the dependent variable from the independent variable to establish that a relationship exists, (b) predict the mediator from the independent variable to establish path a, and (c) predict the dependent variable from both the independent variable and mediator simultaneously to establish path b. More formally, these three steps can be represented by the following regression equations in which \( i_1 \) through \( i_3 \) are the intercepts for the above equations and \( r_1 \) through \( r_3 \) are random disturbances (e.g., error terms):

\[
Y = i_1 + cX + r_1, \quad (1)
\]

\[
M = i_2 + aX + r_2, \quad (2)
\]

\[
Y = i_3 + c'X + bM + r_3. \quad (3)
\]

Here, \( c \) is the total effect of the independent variable on the outcome. The coefficients comprising the indirect effect are the effect of the independent variable on the mediator, \( a \),
and the effect of the mediator on the outcome, $b$. What remains of the independent variable’s influence on the outcome is in $c'$.

We consider only the case of a continuous mediator ($M$) and outcome ($Y$), and briefly discuss other cases at the conclusion of this article. Modern approaches to mediation analysis omit the model in (a), and focus on the paths $a$ and $b$ (e.g., MacKinnon, Lockwood, Hoffman, West, & Sheets, 2002; Rucker, Preacher, Tormala, & Petty, 2011; Shrout & Bolger, 2002). Estimates of path coefficients can be obtained from separate multiple regression models, or estimates for models (b) and (c) can be obtained simultaneously by fitting a structural equation model. In these cases, the covariance between estimates for $\hat{a}$ and $\hat{b}$ is zero (Sobel, 1982, 1986). The paths in the mediational model may also be a part of a hierarchical linear model (or multilevel model; for example, Bauer, Preacher, & Gil, 2006; Kenny, Korchmaros, & Bolger, 2003; Krull & MacKinnon, 2001), or a structural equation model containing latent variables (e.g., Bollen, 1987, 1989; Falk & Biesanz, 2015), which may result in a nonzero covariance between the paths of the indirect effect.

Sobel’s (1982) standard error is based on asymptotic theory, assumes a zero covariance among indirect effect paths, and proposes that the sampling distribution for the indirect effect is normally distributed with the following first-order accurate standard error estimate of the indirect effect: 

$$\sqrt{\hat{a}^2 s_b^2 + \hat{b}^2 s_a^2},$$

where $s_a^2$ and $s_b^2$ are the variance estimates for $\hat{a}$ and $\hat{b}$, respectively. The estimate of indirect effect, $\hat{ab}$, is divided by this standard error to provide a $z$ test. In addition, a 95% CI can be formed by: $\hat{ab} \pm 1.96 \times \sqrt{\hat{a}^2 s_b^2 + \hat{b}^2 s_a^2}$.

In practice, this standard error approximation works well only in very large samples as the sampling distribution of $\hat{ab}$ is often nonnormal, asymmetrical, and depends on the sampling distributions of both $\hat{a}$ and $\hat{b}$. Simulations have shown that Sobel’s standard error yields poorly calibrated CIs and low power to detect indirect effects at sample sizes that are typical of much social science research (e.g., $N \leq 100$; Biesanz et al., 2010; MacKinnon et al., 2002). A modified version of the Baron and Kenny approach that omits Step 1 above and involves just significance tests for $\hat{a}$ and $\hat{b}$ in the mediational model has greater power, good Type I error control, and is theoretically justified (MacKinnon et al., 2002; Rucker et al., 2011). However, this approach requires significance tests for paths $\hat{a}$ and $\hat{b}$ separately, rather than providing a single $p$-value for the indirect effect, $\hat{ab}$, as is provided by Sobel’s test, and does not yield a CI.

The use of resampling methods and the DPR method have emerged as popular modern alternatives to the above classical approaches for making inferences about the indirect effect (Lockwood & MacKinnon, 1998; MacKinnon, Fritz, et al., 2007; MacKinnon, Lockwood, & Williams, 2004; Shrout & Bolger, 2002). These approaches provide a CI for the indirect effect, $\hat{ab}$, and make inferences based on this interval. For instance, if the resulting 95% CI for $\hat{ab}$ does not include 0, a researcher may conclude that the indirect effect is significant at the $\alpha = .05$ level.

In brief, resampling methods such as nonparametric bootstrapping assume that the behavior for a sampling distribution for any statistic can be gleaned from the observed data without making any assumptions about the form of the sampling distribution. Nonparametric bootstrapping begins by obtaining a sample of size $N$ by sampling with replacement from the observed data, where $N$ is the sample size in the original data set. The indirect effect, $\hat{ab}$, is then calculated for this sample and saved. This process repeats a large number of times (e.g., 10,000). Sorting the resulting indirect effect estimates in order, one may obtain a 95% percentile-based (PC) CI for $\hat{ab}$ by calculating the 2.5% and 97.5% quantiles of the resulting distribution. Bias correction and acceleration (BC$a$) theoretically improves the order of accuracy of the obtained CI. Both the PC and BC$a$ bootstrap CIs have been made available to applied researchers through recent SPSS and SAS macros (Preacher & Hayes, 2004; Preacher, Rucker, & Hayes, 2007).

The DPR method starts with the assumption that the sampling distributions for both $\hat{a}$ and $\hat{b}$ are independently normally distributed and well approximated through the use of

![Figure 2](image-url)
the point estimates and standard errors for these paths. The lower and upper 2.5% of the distribution for the indirect effect can then be estimated by using an approximation to the product of two independently normally distributed variables (Meeker & Escobar, 1994). This method of computing CIs for \( \hat{ab} \) has been made widely available through a Windows program written in Fortran that links to SPSS, SAS, and R macros, and a recent R package (MacKinnon, Fritz, et al., 2007; Tofghi & MacKinnon, 2011). However, this method lacks flexibility in situations where the effect of interest is more than just a simple product of two coefficients (e.g., total indirect effect with multiple mediators, dichotomous mediators, or outcomes; Preacher & Selig, 2012).

### Implemented Mediation Analysis Methods

#### The Partial Posterior p-Value: \( p_3 \)

The partial posterior method is based on statistical theory for how to calculate the \( p \)-value for a statistic when its sampling distribution depends on a nuisance parameter (Bayarri & Berger, 1999, 2000). In the case of mediation, the null hypothesis of no mediation, \( H_0: ab = 0 \), is a complex null hypothesis in which the indirect effect can be equal to 0 in the population if either \( a \) or \( b \) is equal to 0. Furthermore, the sampling distribution for the indirect effect depends on both \( a \) and \( b \). Take for an example the case where \( a = 0 \) in the population, meaning the null hypothesis is true. The sampling distribution for \( \hat{ab} \) depends on the unknown population-value for \( b \) (i.e., a nuisance parameter). Smaller values for \( b \) will in general result in a sampling distribution for \( \hat{ab} \) that is more kurtotic and will result in a different \( p \)-value estimate for \( \hat{ab} \) than if a larger value of \( b \) is present in the population. If we wanted to calculate a \( p \)-value for \( \hat{ab} \) under this null hypothesis, we must settle on a value for \( b \) before the sampling distribution for \( \hat{ab} \) is known.

An intuitive choice for \( b \) would be the point estimate \( \hat{b} \) in our sample, and would lead to a \( p \)-value using the plug-in method discussed by Bayarri and Berger (2000) and in the context of mediation analysis by Biesanz et al. (2010),

\[
p_{plug} = \Pr(T > |\hat{ab}| | a = 0, b = \hat{b}),
\]

where \( T \) is some test statistic (in our case, the product \( ab \)). In other words, the partial posterior \( p \)-value is the probability of \( \hat{ab} \), integrating over the partial posterior distribution,

\[
p(b | x, \hat{ab}) \approx \frac{f(x | b)\pi(x)}{f(\hat{ab} | 0, b)},
\]

with the notation \( x / \hat{ab} \) indicating this partialing (following Bayarri & Berger, 2000) and \( f(\hat{ab} | 0, b) \) the density if the observed indirect effect at a point \( b \) (with \( a = 0 \)). The end result is a \( p \)-value for \( \hat{ab} \) under the null hypothesis where \( a = 0 \) in the population. Because it may be more plausible that only \( a \) or \( b \) is 0 in the population, this process repeats for a \( p \)-value for \( \hat{ab} \) under the null hypothesis where \( b = 0 \). To be conservative, the larger of the two \( p \)-values is taken as \( p_3 \)—the final estimate used for making inferences about \( \hat{ab} \).

The algorithm implemented for computing \( p_3 \) is computationally intensive, derived in part from work by Bayarri and Berger (1999, 2000), and based on the following equation when assuming that \( a = 0 \) in the population:

\[
p_3 \approx \frac{\sum_{i=1}^{k} p(\hat{ab} | a = 0, b = b_i)}{\sum_{i=1}^{k} f(\hat{ab} | a = 0, b = b_i)},
\]

where \( k \) represents the number of random draws from the posterior distribution of \( \hat{b} \) (e.g., 1,000,000), with each draw labeled as \( b_i \); the \( p \)-value for \( \hat{ab} \) under that particular draw
is \( p(\hat{a}\hat{b} | a = 0, b = b_h) \) and the density of \( \hat{a}\hat{b} \) under that draw is \( f(\hat{a}\hat{b} | a = 0, b = b_h) \). Thus, for each draw from the posterior of \( \hat{b} \), a \( p \)-value and density of the indirect effect are calculated for that particular value of \( b_h \).

Computationally, it is easier to obtain \( p \)-values and densities for \( \hat{a}\hat{b} \) by using quantities that follow the distribution of the test statistics for \( \hat{a} \) and \( \hat{b} \). Therefore, the software actually implements the following equivalent computational equation when using \( t \)-statistics:

\[
\begin{align*}
\sum_{i=1}^{p} p(t_{f_h}^i | \hat{\lambda}_a = 0, \hat{\lambda}_b = \lambda_h) = p_3 \approx \\
\sum_{i=1}^{p} f(t_{f_h}^i | \hat{\lambda}_a = 0, \hat{\lambda}_b = \lambda_h).
\end{align*}
\]

(7)

Because approximations for the product of two independent \( t \)-distributed variables—one central and one noncentral—are not available, each \( p \)-value and density for \( \hat{a}\hat{b} \) can be determined empirically by multiplying together a large number of random draws (e.g., 1,000,000) from \( \hat{a}/s_a - t(df_a, \hat{\lambda}_a) \) and \( \hat{b}/s_b - t(df_b, \hat{\lambda}_b) \). This algorithm can be easily adapted for use with coefficients from structural equation models or multilevel models by using normal distributions in place of the above \( t \)-distributions. For the remainder of this article, we use \( p_3 \) as short-hand for the partial posterior method using \( t \)-statistics and \( p_{3N} \) as this method using normal approximations.

To avoid having to redo these \( p \)-value and density computations for very similar draws from the posterior of \( \hat{b} \), the program breaks up the posterior distribution draws into a smaller number of equally spaced intervals (e.g., 49 points between \( t_z \pm 5 \)). \( p \)-values and densities for \( \hat{a}\hat{b} \) at each of these points are computed, and splines are used to smooth over these 49 points to obtain \( p \)-value and density estimates for posterior draws that do not fall exactly at these intervals. This approach is similar to conducting Monte Carlo integration over the posterior (i.e., the 1,000,000 draws from the posterior), but using quadrature for estimating \( p \)-values and densities along the posterior distribution. To increase the stability of \( p \)-value estimates, \( p_1 \) may be computed multiple times and the resulting \( p \)-values averaged.

**Hierarchical Bayesian and Monte Carlo CIs**

Hierarchical Bayesian approaches (Biesanz et al., 2010; Huang, Sivaganasen, Succop, & Goodman, 2004) often use Markov chain Monte Carlo methods to determine the posterior distribution of the indirect effect, which can in turn be used to form a CI (for an overview of Bayesian methods in mediation analysis, see also Yuan & MacKinnon, 2009). In our approach, we assume multivariate normally distributed data and simulate from the posterior distribution of each regression coefficient under a noninformative prior (e.g., Berger & Sun, 2008; Gelman, Carlin, Stern, & Rubin, 2004). For instance, draws for the \( a \) coefficient can be simulated from:

\[
\left( a | \hat{a}, s_a \right) \sim \sqrt{z_b} \frac{c_{a,b}}{\sqrt{c_v}} + \hat{a},
\]

(8)

where \( z \) and \( c_{a,b} \) are independent standard normal and central chi-square variates, respectively, with \( v \) degrees of freedom associated with \( s_a \). Because the regression coefficients are assumed independent, a large number of random draws for \( a \) and \( b \) can be multiplied together to form a distribution for the indirect effect. Thus, this method is appropriate when degrees of freedom estimates are available and the covariance among regression coefficients is assumed to be zero. A Bayesian credible interval, which for our purposes is used as a CI, can be formed from the upper \(( 1 - \alpha/2 \) \) and lower \( (\alpha/2 \) \) quantiles of the resulting distribution.

The Monte Carlo method assumes a joint distribution for the estimates \( \hat{a} \) and \( \hat{b} \), such as multivariate normal distribution,

\[
\text{MVN}
\begin{bmatrix}
\hat{a} \\
\hat{b}
\end{bmatrix}
\begin{bmatrix}
s_a^2 & s_{a,b} \\
s_{a,b} & s_b^2
\end{bmatrix}
\]

with a covariance, \( s_{a,b} \), among coefficients (Preacher & Selig, 2012), and can be used when degrees of freedom estimates are not available. In the present context, this distributional assumption is equivalent to that made by the DPR method. In many applications involving multiple regression, the covariance, \( s_{a,b} \), is assumed zero. However, in other contexts, such as mediational models with latent variables, \( s_{a,b} \) may be nonzero (MacKinnon, 2008; see also Tofghi, MacKinnon, & Yoon, 2009) and can be obtained from the inverse information matrix of model parameters as it is with the DPR method (e.g., see Falk & Biesanz, 2015; MacKinnon, 2008). The Monte Carlo method empirically constructs a distribution for \( \hat{a}\hat{b} \) by simulating \( i = 1, 2, \ldots, k \) draws, \( a_i^* \) and \( b_i^* \), from the joint distribution and multiplying each pair of draws together \((a_i^*b_i^*)\). The quantiles of this distribution can be used to form CIs for the indirect effect. Thus, the Monte Carlo method is an empirical approximation to the analytical approach used under the DPR method, but can easily be adapted to handle more complicated situations (Preacher & Selig, 2012).

Both of these methods are similar in that they require draws from a hypothesized distribution for both regression coefficients. The Monte Carlo method in particular is available through an online R calculator (Selig & Preacher, 2008), in recent R packages (Imai, Keele, Tingley, & Yamamoto, 2010; Tofghi & MacKinnon, 2011), code for Stata (Hicks & Tingley, 2011), and macros for SPSS and SAS (Hayes, 2012). Although in most cases a relatively small number of draws (e.g., 1,000) may perform well in simulations, some noticeable instability may be apparent to end users without a very large number of draws. To our knowledge, the above programs and code obtain all draws simultaneously, which can sometimes prohibit the possibility of taking an arbitrarily large number of draws. To circumvent memory (i.e., Random Access Memory) limitations of modern computers and provide
increased stability, we implemented a stochastic approximation algorithm initially presented by Tierney (1983) and as described by Chen and colleagues (Chen et al., 2000). The method was derived by adapting the well-known Robbins-Monro (Robbins & Monro, 1951) stochastic approximation algorithm and we refer readers to these above references for further details. The basic idea is to obtain an initial quantile estimate based on a subset of available data, and revise this estimate as additional data are considered—with decreasing changes in the quantile estimate as more data are collected. We first obtain a fairly large number of draws, \( m = 200,000 \), to obtain initial upper/lower quantile estimates, though note that even a much smaller number of draws could be used in low memory situations. These draws may then be discarded and a new batch of \( m \) draws obtained. The estimate for quantile \( q \) is then updated by:

\[
Q_n = Q_{n-1} + d_{n-1} w_n \left( q - \frac{\#_{i=1}^{m} \{ X_{ni} \leq Q_{n-1} \}}{m} \right).
\]

This process of obtaining \( m \) draws may repeat for a large number of iterations, for example, \( n = 500 \). Thus, \( Q_n \) is the quantile estimate at iteration \( n \), \( X_{ni} \) represents draw \( i \) for iteration \( n \) from the posterior (or sampling) distribution of the indirect effect, \( d_{n-1} = \min(f_{n-1}, f_0^{-\frac{1}{2}} \sqrt{n-1}) \) is the reciprocal of a density estimate of the indirect effect distribution at quantile \( q \) with \( f_{n-1} \) and \( f_0 \) as previous and initial empirical density estimates (also updated stochastically), \( \#_{i=1}^{m} \{ X_{ni} \leq Q_{n-1} \} \) represents the number of draws at the current iteration that are less than or equal to the previous quantile estimate, and \( w_n = 1/n \). Note that the weight \( w_n \) ensures that later iterations have a diminishing effect on the quantile estimate, and choosing \( \min(f_{n-1}, f_0^{-\frac{1}{2}} \sqrt{n-1}) \) as the estimate for \( d_{n-1} \) ensures that this quantity does not become too large. This method performs almost as well as if all \( m \times n \) draws from the posterior (or sampling) distribution were to be maintained in memory and is asymptotically equivalent.

**Software Implementation**

The programs that implement the computational algorithms are written in Java and makes use of the open-source statistical library SSJ version 2.4 (L’Ecuyer, 2010; L’Ecuyer & Buist, 2005), ensuring that any computer with an updated version of Java (8 or later) should be able to run the programs. Some features, such as obtaining a large number of random draws and performing additional computations with them (e.g., multiplying draws together), can be done independently and are easily parallelized. Therefore, the programs also utilize java.util.concurrent to use all free processing cores and for a faster overall computational time with multicore CPUs. As of writing this manuscript, the program has been tested on Windows 7, Ubuntu Linux 14.04, and Mac OS 10.7.5. The program and source code can be accessed at the following website: https://www.msu.edu/~falkcarl/mediation.html or by contacting the authors of this manuscript. The programs also use a graphical interface that may be easier to use for some researchers than editing of SPSS, SAS, and R code or macros. We describe the interface in the empirical example at the end of this manuscript.

To use the programs, researchers need only know how to obtain point estimates and standard errors for the indirect effect estimates, compute test statistics for \( \hat{a} \) and \( \hat{b} \) (e.g., \( t_a = \hat{a} / \hat{\sigma}_a \) and \( t_b = \hat{b} / \hat{\sigma}_b \)), and in some cases their associated degrees of freedom, \( df_a \) and \( df_b \). This involves estimates of the coefficients in Equations 2 and 3 (see also Figure 2) using any available statistical package. Researchers familiar with traditional approaches to mediation analysis such as Baron and Kenny’s (1986) approach and Sobel’s test (Sobel, 1982) already know how to obtain these values, and such values would typically be required for publication of any manuscript that reports mediation analysis. When degrees of freedom estimates are not available (e.g., when using a structural equation model), normal approximations that use \( P_{SN} \) and \( CI_{MC} \) may be used. When the paths \( \hat{a} \) and \( \hat{b} \) are not independent, as in the case of structural equation models with latent variables and particular types of hierarchical linear models (see Bauer et al., 2006; Kenny et al., 2003), the correlation among regression coefficients required for \( CI_{MC} \) may then be obtained from the covariance matrix of the regression coefficients (see MacKinnon, 2008).

Some user control over the computational speed and accuracy of the programs is offered through two settings that we later refer to as the “Good” and “Excellent” settings. These settings refer to the number of repetitions for \( P_3 \) and \( P_{SN} \) (1 and 5) and number of iterations for \( CI_{HB} \) and \( CI_{MC} \) (100 and 500) as described in the technical details of each method and exactly as tested under the simulations we present in the following section. Like bootstrapping, these methods are computationally intensive and some random variation may occur from run to run, even with the same input values. Thus, although all settings perform well in simulations, users may notice that the “Excellent” setting takes longer to compute but varies less from run to run.

To illustrate, computation of \( P_3 \) and \( CI_{HB} \) was repeated each with four different settings (1, 2, 5, and 10 repetitions for \( P_3 \) and 100, 200, 500, and 1,000 iterations for \( CI_{HB} \)) and done 1,000 times on the data presented later as an illustrative example. Because the input values did not change, any variation in the estimates is due to random variation across different runs of the program. The distribution of \( p \)-values from \( P_3 \) appears in Figure 3 and the lower and upper CI estimates from \( CI_{HB} \) appear in Figures 4 and 5. Note first that the least intensive setting already yields values that differ relatively little from run to run. For example, \( P_3 \) estimates with 1 repetition (the “Good” setting) range from .028 to .030—a difference of only .002. Note also that the most computationally intensive setting (10 and 1,000 for \( P_3 \) and \( CI_{HB} \), respectively) results in a tighter distribution of \( p \)-values and endpoints of each interval, but appears to be only slightly
better than the next lowest setting (5 and 500, the “Excellent” settings) while requiring double the amount of computational time. Therefore, we recommend the “Good” setting for general use, and the “Excellent” setting when the p-value or CI endpoints are near an important boundary condition or when conducting final estimation before publication.

Simulations

We now briefly present a simulation study. The primary purpose of simulations was to check that the underlying code for the programs performed as expected against the popular PC and BC\alpha nonparametric bootstrap methods under a novel set of conditions that should conceptually replicate previous research (Biesanz et al., 2010). In addition, we are unaware of any previous research directly comparing the CIHB and CIMC approaches and how the normal approximation for \( p_{3N} \) performs relative to \( p_3 \).

Design

Data were generated from multivariate normal distributions with all variables having unit variance. Ten different effect size conditions were used with four to evaluate Type I errors (\( \alpha = 0.01, 0.05 \), all with \( p = 0 \)), and six to evaluate Power and CI coverage rates (\( \alpha = 0.05 \) fully crossed with \( b = 0, 0.05 \)). One-thousand data sets per condition were generated, each with \( N = 75 \) observations. Data generation was conducted in R (Version 2.12.0) using the MASS package (R Development Core Team, 2011; Venables & Ripley, 2002) and included only multivariate normal data.

In each cell of the design, PC and BC\alpha bootstrap intervals at \( \alpha = 0.05 \) and \( \alpha = 0.01 \) were estimated based on 12,500 resamples using the boot package (Canty & Ripley, 2011). CIHB, CI\alpha, \( p_3 \), and \( p_{3N} \) were estimated as described in the technical details sections above and implemented in Java using open-source statistical library SSJ version 2.4 (L’Ecuyer, 2010; L’Ecuyer & Buist, 2005) and compiled using JDK Version 1.6.0 (or Java 6.0). These methods were computed twice using different settings for the stability of each algorithm; specifically, 100 and 500 iterations were used for CIs, and the \( p_3 \) and \( p_{3N} \) methods were repeated 1 and 5 times with the resulting p-values averaged in the latter condition. Most simulations were conducted on a WestGrid cluster of 1,680 3.06 GHz processors with the exception of CI\alpha and \( p_{3N} \) methods, which were run on a desktop computer with an Intel Core2 i7-3770 3.40 GHz CPU.
Table 1. Type I Error Rates for All Methods.

| Nominal rate | Paths | PC | BC_a | p_3 (1) | p_3 (5) | p_3N (1) | p_3N (5) | CI_{HB} (100) | CI_{HB} (500) | CI_{MC} (100) | CI_{MC} (500) |
|--------------|-------|----|------|---------|---------|----------|----------|---------------|---------------|---------------|---------------|
| α = .05      | a = .0, b = .0 | .001 | .005 | .002 | .002 | .002 | .002 | .001 | .001 | .001 | .001 |
|              | a = .1, b = .0 | .006 | .020 | .011 | .011 | .011 | .011 | .006 | .006 | .006 | .006 |
|              | a = .3, b = .0 | .031 | .070 | .045 | .045 | .049 | .049 | .025 | .026 | .028 | .028 |
|              | a = .5, b = .0 | .051 | .074 | .055 | .054 | .054 | .054 | .042 | .043 | .046 | .046 |
| α = .01      | a = .0, b = .0 | .000 | .000 | .000 | .000 | .000 | .000 | .000 | .000 | .000 | .000 |
|              | a = .1, b = .0 | .001 | .001 | .000 | .000 | .000 | .000 | .000 | .000 | .000 | .000 |
|              | a = .3, b = .0 | .004 | .012 | .006 | .006 | .005 | .007 | .004 | .004 | .005 | .005 |
|              | a = .5, b = .0 | .015 | .021 | .018 | .018 | .018 | .018 | .016 | .016 | .017 | .017 |

Note. Values above Serlin’s criterion appear in bold. Values below Serlin’s criterion are italicized. PC = percentile bootstrap; BC_a = bias-corrected and accelerated bootstrap; p_3 (1) and p_3 (5) = partial posterior with 1 and 5 repetitions, respectively; CI_{HB} (100) and CI_{HB} (500) = hierarchical Bayesian confidence intervals with 100 and 500 iterations, respectively.

Results

Type I error. Type I error rates at α = .05 and α = .01 appear in Table 1. Thus, methods with good performance are expected to reject the null hypothesis of no mediation a proportion of .05 and .01 in each of these conditions, respectively, with higher rejection arguably worse than low rejection rates. To highlight values that fall outside of Serlin’s (2000) criterion, values with low Type I error rates (<.0375 or <.006 for α = .05 and α = .01, respectively) are italicized and values with high Type I error rates (>-.0625 or >.015 for α = .05 and α = .01, respectively) appear in bold. First note that there is virtually no difference across the two different settings used for computational precision: 100 versus 500 iterations for CI_{HB} and CI_{MC}, and 1 or 5 repetitions for p_3 and p_{3N}. We also note that under the simulated conditions, there were negligible differences between CI_{HB} and CI_{MC}, and between p_3 and p_{3N} with Type I error rates differing by only .004 at most under α = .05. The results indicated that all methods tend to underreject when the a path is zero or small (.1), with the largest rejection rate being .02 when α = .05 and .001 when α = .01. When the a path is .3 or .5, however, some overrejection (i.e., high Type I error) is apparent. The BC_a method overrejection in two cells (reaching .07 and .074) when α = .05, whereas all other methods maintained good Type I error rates with p_{3N} being close to .05 when a = .3. When α = .01 and a = .3, p_3, p_{3N}, and BC_a maintained proper Type I error rates while all other methods underrejection. Finally, when α = .01 and a = .5, all methods had a tendency to overreject with BC_a, having the highest Type I error (.021) and PC having the lowest (.015).

Power. Power for all methods in all cells of the design appears in Table 2. We note that although the BC_a method tended to have the highest power, choice of this method based on power may be misguided when there is a risk of inflated Type I error rates under some conditions. Following the BC_a method, p_{3N} tended to have second highest power across all cells of the simulation design. The gains in power by using normal approximations, CI_{MC} and p_{3N}, tended to be greater than the corresponding inflation in Type I error. For instance, p_{3N} had power that was .019 greater than p_3 when α = .05 and both paths were .3. In general, the PC method had a tendency to have slightly higher power than both CI_{HB} and CI_{MC}, but this was not uniformly the case (e.g., α = .05 when a = .3 or .5 and b = .5) and typically power did not differ by much across these methods. For example, the largest difference between PC and CI_{HB} occurred when a = .3 and b = .3 and was .02 when α = .05 and .044 when α = .01.

CI coverage rates. Table 3 shows 95% and 99% CI coverage rates, in which we would expect methods with good coverage to contain the true parameter a proportion of .95 and .99 of the time in these conditions. To highlight values that fall outside of Serlin’s (2000) criterion, values with low coverage rates (<.9375 or <.986 for α = .05 and α = .01, respectively) are in bold and values with high coverage rates (>-.9625 or >.994 for α = .05 and α = .01, respectively) are italicized. Arguably, low coverage rates are worse than high coverage, as this would mean that the true indirect effect falls outside of the CI more than what is expected. By these standards, clearly CI_{HB} and CI_{MC} had the best coverage rates, only experiencing coverage rates that were low in a single cell in the design (α = .01 when a = .5 and b = .5) and displayed results that were nearly identical to each other. By contrast, other approaches performed worse with PC having three cells with low coverage and the BC_a method experiencing low coverage in six cells.

Discussion

Overall, the results presented here are consistent with previous research (Biesanz et al., 2010) and indicate that the underlying programs’ code is operating as expected. For instance, the results are consistent with previous research indicating that the BC_a bootstrap has inflated Type I error in some cells of the experimental design, while other methods have lower Type I error rates. p_3 and p_{3N} also typically had higher power than methods other than the BC_a, with the remaining methods not differing by much except in a few...
cells in the design. The BCa method’s poorly calibrated CIs (i.e., coverage rates) are also consistent with previous research. One novel finding was that the CIHB and CIMC had better CI coverage rates than the PC method in the conditions studied whereas typically these methods have been found to perform similarly. Finally, at the sample sizes used in this simulation, there was little difference among PC and CIHB versus analogous methods that use the same computational approach and make the use of normal approximations, $p_{3N}$ and CIMC, though in some cases the use of normal approximations resulted in higher power and sometimes slightly higher Type I error rates. We would expect a greater divergence among these methods at smaller sample sizes as the normal approximation becomes less reasonable, and we suspect that the pattern of gains in Type I error and power may be a complicated function of the method used to make inferences, sample size, and the magnitude of the indirect effect (e.g., Fritz, Taylor, & MacKinnon, 2012).

### Illustrative Example

To provide a concrete example, in Dunn and colleagues’ (2007) Study 2B, it was expected that instructing romantic partners to self-present or not ($X$) would lead to changes in actual self-presentation as coded by independent raters ($M$) which would in turn lead to changes in participants’ self-reported positive affect ($Y$). To test the first link, a model equivalent to Equation 2 by predicting $M$ from $X$ was fit to the data, revealing that those explicitly asked to self-present displayed higher self-presentation levels, $a = 1.13$, $t(38) = 3.165$ (with a standard error estimate of $s_a = .357$). These values are then input into the programs (Figures 6 and 7 for

| Table 2. Power for All Methods. |
|----------------------------------|
| Nominal rate | Paths | PC | BCa | $p_{3N}$ (1) | $p_{3N}$ (5) | CIHB (100) | CIHB (500) | CIMC (100) | CIMC (500) |
| $\alpha = .05$ | $a = .1, b = .3$ | .100 | .154 | .110 | .110 | .115 | .116 | .087 | .087 | .090 | .090 |
| | $a = .3, b = .3$ | .517 | .636 | .582 | .582 | .603 | .603 | .497 | .497 | .518 | .519 |
| | $a = .5, b = .3$ | .626 | .674 | .654 | .656 | .657 | .657 | .661 | .661 | .623 | .627 |
| | $a = .1, b = .5$ | .153 | .196 | .167 | .164 | .168 | .169 | .139 | .139 | .145 | .145 |
| | $a = .3, b = .5$ | .752 | .804 | .790 | .792 | .794 | .795 | .756 | .756 | .773 | .773 |
| | $a = .5, b = .5$ | .975 | .985 | .984 | .984 | .985 | .984 | .976 | .976 | .979 | .979 |
| $\alpha = .01$ | $a = .1, b = .3$ | .035 | .057 | .035 | .032 | .038 | .038 | .024 | .024 | .029 | .029 |
| | $a = .3, b = .3$ | .253 | .340 | .276 | .274 | .294 | .293 | .210 | .210 | .243 | .245 |
| | $a = .5, b = .3$ | .384 | .452 | .404 | .401 | .419 | .415 | .352 | .352 | .382 | .383 |
| | $a = .1, b = .5$ | .052 | .062 | .050 | .049 | .054 | .053 | .038 | .038 | .044 | .045 |
| | $a = .3, b = .5$ | .505 | .570 | .530 | .536 | .552 | .551 | .494 | .494 | .515 | .517 |
| | $a = .5, b = .5$ | .905 | .932 | .922 | .924 | .927 | .926 | .899 | .899 | .916 | .916 |

Note. PC = percentile bootstrap; BCa = bias-corrected and accelerated bootstrap; $p_{3N}$ (1) and $p_{3N}$ (5) = partial posterior with 1 and 5 repetitions, respectively; CIHB (100) and CIHB (500) = hierarchical Bayesian confidence intervals with 100 and 500 iterations, respectively.

| Table 3. CI Coverage Rates for All Methods That Yield CIs. |
|----------------------------------|
| Nominal rate | Paths | PC | BCa | (100) | (500) | (100) | (500) |
| $\alpha = .05$ | $a = .1, b = .3$ | .941 | .908 | .948 | .948 | .946 | .946 |
| | $a = .3, b = .3$ | .940 | .952 | .949 | .949 | .947 | .947 |
| | $a = .5, b = .3$ | .948 | .944 | .954 | .954 | .953 | .953 |
| | $a = .1, b = .5$ | .936 | .922 | .941 | .941 | .939 | .939 |
| | $a = .3, b = .5$ | .942 | .949 | .957 | .957 | .952 | .952 |
| | $a = .5, b = .5$ | .943 | .952 | .952 | .952 | .949 | .949 |
| $\alpha = .01$ | $a = .1, b = .3$ | .991 | .970 | .993 | .993 | .991 | .991 |
| | $a = .3, b = .3$ | .989 | .988 | .992 | .992 | .989 | .989 |
| | $a = .5, b = .3$ | .986 | .983 | .989 | .989 | .989 | .989 |
| | $a = .1, b = .5$ | .982 | .974 | .987 | .987 | .986 | .986 |
| | $a = .3, b = .5$ | .992 | .995 | .995 | .995 | .995 | .995 |
| | $a = .5, b = .5$ | .983 | .983 | .985 | .985 | .984 | .984 |

Note. Values below Serlin’s criterion appear in bold. Values above Serlin’s criterion are italicized. CI = confidence interval; PC = percentile bootstrap; BCa = bias-corrected and accelerated bootstrap; CIHB (100) and CIHB (500) = hierarchical Bayesian confidence intervals with 100 and 500 iterations, respectively.
Note. CIHB = Hierarchical Bayes confidence interval.

$p_3$ and CIHB programs, respectively). Note also that inputting these values with more decimal places affords more accurate estimates. Then, a model equivalent to Equation 3 found that self-presentation led to higher levels of positive affect, $b = 0.19, t(37) = 2.153$ (with a standard error estimate of $s_b = .088$), and these values are also used as input into the program (Figures 6 and 7 for $p_3$ and CIHB programs, respectively). Users have the option of changing the computational accuracy/speed of each program, as described in the Technical and Computational Details section above, by selecting “Good” or “Excellent.” If $p_{3N}$ or CIHM are desired, the users may select a different “Computational Method” labeled as “Normal approximation.” The user then needs only to click the “Compute” button, and wait for output from the program. Progress for computation is tracked via a progress bar before the result is displayed, and in this case the indirect effect is significant, $ab = .21, p_3 = .03, 95\% \text{ CI}_{\text{HB}} = [.01, .50]$ (Figures 8 and 9 for $p_3$ and CIHB programs, respectively). Overall, this suggests a dichotomous decision in favor of the indirect effect. That is, actual self-presentation (independently rated) can at least partly explain the effect of a self-presentation manipulation on positive affect. Although CIs for the indirect effect in this case do not necessarily suggest a high level of precision in the effect estimate, with the lower bound near zero. On a desktop computer with an Intel Core2 i7-3770, 3.40Gz processor, these computations under the “Good” setting took approximately 48s and 10s for $p_3$ and CIHB, respectively. Note that if the Monte Carlo (Normal approximation) method is used for CIs, the corresponding program shades out input corresponding to $df$ and allows the user to input a value for the correlation among paths (see Figure 10).

**Discussion and Conclusion**

To keep up with the most advanced statistical methods, it is vital that applied researchers are given access to these methods...
In conclusion, reliance on traditional methods (e.g., Sobel’s test) likely results in many indirect effects that go undetected due to statistical power that is too low. We hope that the availability of $p_3$ and $P_{SN}$ will help reduce the reliance these traditional methods while offering higher power and the same simplicity in computation and presentation. Likewise, use of bootstrapping relies on access to the raw data; in the case of the BC$_3$ bootstrap, this may also result in high Type I error rates and poorly calibrated CIs. The availability of $CI_{HB}$ and $CI_{MC}$ provides researchers with a flexible software solution to computing CIs that perform well in controlling Type I error and are very well calibrated.

**Declaration of Conflicting Interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research and/or authorship of this article: This study was enabled in part by support provided by WestGrid (www.westgrid.ca) and Compute Canada Canada (www.computecanada.ca), and the Social Sciences and Humanities Research Council of Canada (410-2011-1962 and 435-2014-1558).

**Notes**

1. Note that the distribution of the product (DPR) method used by *Rmediation* is referred to by Biesanz, Falk, and Savaei (2010) as a “revised” DPR method that produces a different quantity than the R macro Prodclin.r, which was formerly available from http://www.public.asu.edu/~davidpm/ripl/Prodclin/.
2. The posterior distribution is given in Equation 8 in the section “Hierarchical Bayesian and Monte Carlo CIs.”
3. Note that when the null hypothesis is true in structural equation models, there is no covariance between $\hat{\gamma}$ and $\hat{\delta}$, making it unnecessary to consider this covariance for computation of $P_{SN}$ as this is a $p$-value under the null.
4. Berger and Sun (2008) provide details of the prior and posterior in the case of bivariate regression. The posterior distribution listed here is a generalization that also holds in the case of multiple regression.
5. Both programs are freely distributable and modifiable under the GNU General Public License (v3). Future versions of the program will also be available by contacting the authors and via the website: www.appliedregression.com.
6. The covariance matrix among regression coefficients in structural equation models is the same as the inverse information matrix of model parameters. For example, if using the lavaan package in R (Rosseel, 2012), the “vcov” function can be used to extract the relevant matrix and “cov2cor” will standardize this matrix. In Mplus (Muthén & Muthén, 1998-2010), the standardized matrix can be obtained by requesting “TECH3” output. Once obtained, this standardized matrix is obtained, the corresponding value for the correlation between paths can be used in computation of $CI_{MC}$.
7. The number of resamples here takes approximately the same computational time as 500 iterations of the $CI_{HB}$ method if both are conducted in R and without parallel processing.
References

Baron, R. M., & Kenny, D. A. (1986). The moderator-mediator variable distinction in social psychological research: Conceptual, strategic and statistical considerations. *Journal of Personality and Social Psychology, 51*, 1173-1182.

Bauer, D. J., Preacher, K. J., & Gil, K. M. (2006). Conceptualizing and testing random indirect effects and moderated mediation in multilevel models: New procedures and recommendations. *Psychological Methods, 11*, 142-163.

Bayarri, M. J., & Berger, J. O. (1999). Quantifying surprise in the data and model verification [with discussion]. In J. M. Bernardo, J. O. Berger, A. P. Dawid, & A. F. M. Smith (Eds.), *Bayesian statistics 6* (pp. 53-82). Oxford, UK: Oxford University Press.

Bayarri, M. J., & Berger, J. O. (2000). P-values for composite null models [with discussion]. *Journal of the American Statistical Association, 95*, 1127-1142.

Berger, J. O., & Sun, D. (2008). Objective priors for the bivariate normal model. *The Annals of Statistics, 36*, 963-982.

Biesanz, J. C., Falk, C. F., & Savalei, V. (2010). Assessing mediational models: Testing and interval estimation for indirect effects. *Multivariate Behavioral Research, 45*, 661-701.

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. *Sociological Methodology, 17*, 37-69.

Bollen, K. A. (1989). *Structural equations with latent variables*. New York, NY: John Wiley.

Canty, A., & Ripley, B. (2011). boot: Bootstrap R (S-Plus) functions (R package version 1.3-4). Retrieved from http://cran.r-project.org/package=boot

Chen, F., Lambert, D., & Pinheiro, J. C. (2000). Incremental quantile estimation for massive tracking. In *Proceedings of the Sixth Association for Computing Machinery’s Special Interest Group on Knowledge Discovery and Data Mining (ACM SIGKDD) International Conference on Knowledge Discovery and Data Mining* (pp. 516-522). New York, NY: ACM Press.

Dunn, E. W., Biesanz, J. C., Human, L. J., & Finn, S. (2007). Misunderstanding the affective consequences of everyday social interactions: The hidden benefits of putting one’s best face forward. *Journal of Personality and Social Psychology, 92*, 990-1005.

Falk, C. F., & Biesanz, J. C. (2015). Inference and interval estimation methods for indirect effects with latent variable models. *Structural Equation Modeling, 22*, 24-38.

Fritz, M. S., Taylor, A. B., & MacKinnon, D. P. (2012). Explanation of two anomalous results in statistical mediation analysis. *Multivariate Behavioral Research, 47*, 61-87.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian data analysis*. New York, NY: Chapman & Hall.

Hayes, A. F. (2012). *PROCESS: A versatile computational tool for observed variable mediation, moderation, and conditional process modeling* [White paper]. Retrieved from http://www.afhayes.com/public/process2012.pdf

Hicks, R., & Tingley, D. (2011). Causal mediation analysis. *Stata Journal, 11*, 609-615.

Huang, B., Sivaganasen, S., Succop, P., & Goodman, E. (2004). Statistical assessment of mediational effects for logistic mediational models. *Statistics in Medicine, 23*, 2713-2728.

Iacobucci, D. (2012). Mediation analysis and categorical variables: The final frontier. *Journal of Consumer Psychology, 22*, 582-594.

Imai, K., Keele, L., & Tingley, D. (2010). A general approach to causal mediation analysis. *Psychological Methods, 15*, 309-334.

Imai, K., Keele, L., Tingley, D., & Yamamoto, T. (2010). Causal mediation analysis using R. In H. D. Vinod (Ed.), *Advances in social science research using R* (pp. 129-154). New York, NY: Springer.

Kenny, D. A., Korchmaros, J. D., & Bolger, N. (2003). Lower level mediation in multilevel models. *Psychological Methods, 8*, 115-128.

Krull, J. L., & MacKinnon, D. P. (2001). Multilevel modeling of individual and group level mediated effects. *Multivariate Behavioral Research, 36*, 249-277.

L’Ecuyer, P. (2010). SSJ: Stochastic Simulation in Java (Version 2.4), Département d’Informatique Recherche Opérationnelle (DIRO), Université de Montréal. Retrieved from http://www.iro.umontreal.ca/~simardr/ssj/indexe.html

L’Ecuyer, P., & Buist, E. (2005, December 4-7). Simulation in Java with SSJ. In M. E. Kuhl, N. M. Steiger, F. B. Armstrong, & J. A. Joines (Eds.), *Proceedings of the 2005 Winter Simulation Conference* (pp. 611-620). Retrieved from http://dx.doi.org/10.1109/WSC.2005.1574301

Lockwood, C., & MacKinnon, D. P. (1998). Bootstrapping the standard error of the mediated effect. In *Proceedings of the 23rd annual meeting of SAS Users Group International* (pp. 997-1002). Cary, NC: SAS Institute Inc.

MacKinnon, D. P. (2008). *Introduction to statistical mediation analysis*. New York, NY: Routledge Academic.

MacKinnon, D. P., Fairchild, A. J., & Fritz, M. S. (2007). Mediation analysis. *Annual Review of Psychology, 58*, 593-614.

MacKinnon, D. P., Fritz, M. S., Williams, J., & Lockwood, C. M. (2007). Distribution of the product confidence limits for the indirect effect: Program PRODCLIN. *Behavior Research Methods, Instruments, & Computers, 39*, 384-389.

MacKinnon, D. P., Lockwood, C. M., Hoffman, J. M., West, S. G., & Sheets, V. (2002). A comparison of methods to test mediation and other intervening variable effects. *Psychological Methods, 7*, 83-104.

MacKinnon, D. P., Lockwood, C. M., & Williams, J. (2004). Confidence limits for the indirect effect: Distribution of the product and resampling methods. *Multivariate Behavioral Research, 39*, 99-128.

Meeker, W. Q., & Escobar, L. A. (1994). An algorithm to compute the CDF of the product of two normal random variables. *Communications in Statistics: Simulation and Computation, 23*, 271-280.

Muthén, L. K., & Muthén, B. O. (1998-2010). *Mplus user’s guide* (6th ed.). Los Angeles, CA: Author.

Nosek, B. A., Graham, J., Lindner, N. M., Kesebir, S., Hawkins, C. B., Hahn, C., . . . Tenney, E. R. (2010). Cumulative and career-stage citation impact of social-personality psychology programs and their members. *Personality and Social Psychology Bulletin, 36*, 1283-1300. Retrieved from http://project Implicit.net/nosek/papers/citations/citedarticles.htm

Preacher, K. J., & Hayes, A. F. (2004). *SPSS and SAS procedures for estimating indirect effects in simple mediation models*. *Behavior Research Methods, Instruments, & Computers, 36*, 717-731.

Preacher, K. J., Rucker, D. D., & Hayes, A. F. (2007). Addressing moderated mediation hypotheses, theory, methods, and prescriptions. *Multivariate Behavioral Research, 42*, 185-227.

Preacher, K. J., & Selig, J. P. (2012). Advantages of Monte Carlo confidence intervals for indirect effects. *Communication Methods and Measures, 6*, 77-98.
R Development Core Team. (2011). *R: A language and environment for statistical computing*. Vienna, Austria: R foundation for Statistical Computing. Available from http://www.R-project.org/

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics, 22*, 400-407.

Rosseel, Y. (2012). Lavaan: An R package for structural equation modeling. *Journal of Statistical Software, 48*, 1-36.

Rucker, D. D., Preacher, K. J., Tormala, Z. L., & Petty, R. E. (2011). Mediation analysis in social psychology: Current practices and new recommendations. *Social & Personality Psychology Compass, 5*, 359-371.

Selig, J. P., & Preacher, K. J. (2008). Monte Carlo method for assessing mediation: An interactive tool for creating confidence intervals for indirect effects [Computer software]. Available from http://quantpsy.org/

Serlin, R. C. (2000). Testing for robustness in Monte Carlo studies. *Psychological Methods, 5*, 230-240.

Shrout, P. E., & Bolger, N. (2002). Mediation in experimental and nonexperimental studies: New procedures and recommendations. *Psychological Methods, 7*, 422-445.

Sobel, M. E. (1982). Asymptotic confidence intervals for indirect effects in structural equation models. *Sociological Methodology, 13*, 290-312.

Sobel, M. E. (1986). Some new results on indirect effects and their standard errors in covariance structure models. *Sociological Methodology, 16*, 159-186.

Tierney, L. (1983). A space-efficient recursive procedure for estimating a quantile of an unknown distribution. *SIAM Journal on Scientific and Statistical Computing, 4*, 706-711.

Tofighi, D., & MacKinnon, D. P. (2011). RMediation: An R package for mediation analysis confidence intervals. *Behavior Research Methods, Instruments, & Computers, 43*, 692-700.

Tofighi, D., MacKinnon, D. P., & Yoon, M. (2009). Covariances between regression coefficient estimates in a single mediator model. *British Journal of Mathematical and Statistical Psychology, 62*, 457-484.

Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with S* (4th ed.). New York, NY: Springer.

Yuan, Y., & MacKinnon, D. P. (2009). Bayesian mediation analysis. *Psychological Methods, 14*, 301-322.

Yuan, Y., & MacKinnon, D. P. (2014). Robust mediation analysis based on median regression. *Psychological Methods, 19*, 1-20.

**Author Biographies**

**Carl F. Falk** is an assistant professor of Measurement and Quantitative Methods in the College of Education at Michigan State University. His current research focuses on the development, testing, and computer programming of latent variable models with applications across the social sciences.

**Jeremy C. Biesanz** is an associate professor in the Department of Psychology at the University of British Columbia.