

**UCLA**

**UCLA Electronic Theses and Dissertations**

**Title**

Model-Based Imputation for Multilevel Interaction Effects

**Permalink**

<https://escholarship.org/uc/item/0mh203nn>

**Author**

Keller, Brian Tinnell

**Publication Date**

2019

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA  
Los Angeles

Model-Based Imputation for Multilevel Interaction Effects

A dissertation submitted in partial satisfaction  
of the requirements for the degree  
Doctor of Philosophy in Psychology

by

Brian Tinnell Keller

2019

© Copyright by  
Brian Tinnell Keller  
2019

# ABSTRACT OF THE DISSERTATION

Model-Based Imputation for Multilevel Interaction Effects

by

Brian Tinnell Keller

Doctor of Philosophy in Psychology

University of California, Los Angeles, 2019

Professor Craig K. Enders, Chair

Over the last few decades, a large body of research supports the use of multiple imputation as a method for handling missing data. Despite imputation's broad appeal, the method is known to introduce biases when applied to models with interactive and polynomial effects. In the context of single-level regression models, multiple imputation based on a fully Bayesian model specification has shown great promise, but limited research to date has considered this approach for multilevel models. The purpose of this dissertation is to investigate the multilevel extension of Bayesian model-based imputation to a two-level regression model with a cross-level interactive effect. With the exception of some rather extreme scenarios with non-normal data, computer simulations from this research suggest that the model-based approach can effectively estimate these models in a wide variety of conditions that are typical of social and behavioral science research data. In virtually every condition examined, model-based imputation outperformed existing alternatives to handling incomplete interactive effects. This procedure is available in the Blimp software package for macOS, Windows, and Linux.

The dissertation of Brian Tinnell Keller is approved.

Stephen G. West

Jennifer L. Krull

Martin M. Monti

Han Du

Craig K. Enders, Committee Chair

University of California, Los Angeles

2019

*To my family . . .  
who—among so many other things—  
supported me throughout this endeavor.*

## TABLE OF CONTENTS

<b>List of Figures</b> . . . . .	<b>viii</b>
<b>List of Tables</b> . . . . .	<b>x</b>
<b>Acknowledgments</b> . . . . .	<b>xi</b>
<b>Vita</b> . . . . .	<b>xii</b>
<b>1 Introduction</b> . . . . .	<b>1</b>
1.1 Missing Data Mechanisms . . . . .	2
1.2 Introduction to Multiple Imputation . . . . .	3
1.3 Bayesian Estimation of Two-level Model . . . . .	4
1.4 FCS Imputation of Level-1 Incomplete Variable . . . . .	6
1.5 FCS Imputation of Level-2 Incomplete Variable . . . . .	8
<b>2 Literature Review</b> . . . . .	<b>10</b>
2.1 Imputation of Single-level Interaction Effects . . . . .	10
2.2 Model-Based Imputation . . . . .	14
2.3 Multilevel Model-Based Imputation . . . . .	22
<b>3 Methods</b> . . . . .	<b>27</b>
3.1 Analysis Model . . . . .	28
3.2 Overview of Simulation Conditions . . . . .	28
3.3 Data Generation . . . . .	30
3.3.1 Covariate model for Simulation 2 . . . . .	32

3.3.2	Covariate model for Simulation 3 . . . . .	34
3.3.3	Covariate model for Simulation 4 . . . . .	35
3.3.4	Covariate model for Simulation 5 . . . . .	37
3.4	Simulation Implementation and Outcome Measures . . . . .	38
<b>4</b>	<b>Results . . . . .</b>	<b>40</b>
4.1	Simulation 1 – Normally Distributed Covariates . . . . .	40
4.2	Simulation 2 – Level-2 Binary Covariate . . . . .	42
4.3	Simulation 3 – Non-normal Level-2 Covariate . . . . .	44
4.4	Simulation 4 – Level-1 Covariate with Non-normal Level-2 Residual . . . . .	46
4.5	Simulation 5 – Level-1 Covariate with Non-normal Level-1 Residual . . . . .	48
<b>5</b>	<b>Discussion . . . . .</b>	<b>52</b>
5.1	Summary of Findings . . . . .	53
5.2	Limitations and Future Research . . . . .	55
5.3	Conclusion . . . . .	57
<b>A</b>	<b>Figures . . . . .</b>	<b>58</b>
<b>B</b>	<b>Simulation 1: 15% Missing Data Rate Results . . . . .</b>	<b>84</b>
<b>C</b>	<b>Simulation 2: 15% Missing Data Rate Results . . . . .</b>	<b>87</b>
<b>D</b>	<b>Simulation 3: 15% Missing Data Rate Results . . . . .</b>	<b>92</b>
<b>E</b>	<b>Simulation 4: 15% Missing Data Rate Results . . . . .</b>	<b>95</b>
<b>F</b>	<b>Simulation 5: 15% Missing Data Rate Results . . . . .</b>	<b>98</b>

<b>G</b>	<b>Simulation 1: Listwise Deletion Results . . . . .</b>	<b>101</b>
<b>H</b>	<b>Simulation 2: Listwise Deletion Results . . . . .</b>	<b>104</b>
<b>I</b>	<b>Simulation 3: Listwise Deletion Results . . . . .</b>	<b>109</b>
<b>J</b>	<b>Simulation 4: Listwise Deletion Results . . . . .</b>	<b>112</b>
<b>K</b>	<b>Simulation 5: Listwise Deletion Results . . . . .</b>	<b>115</b>
	<b>References . . . . .</b>	<b>117</b>

## LIST OF FIGURES

1	Simulation 1: Percent Bias – ICC = 0.10; Missing = 30% . . . . .	59
2	Simulation 1: Percent Bias – ICC = 0.50; Missing = 30% . . . . .	60
3	Simulation 1: Coverage – ICC = 0.10; Missing = 30% . . . . .	61
4	Simulation 1: Coverage – ICC = 0.50; Missing = 30% . . . . .	62
5	Simulation 2: Percent Bias – ICC = 0.10; Missing = 30%; Split = 50/50 . . . . .	63
6	Simulation 2: Percent Bias – ICC = 0.10; Missing = 30%; Split = 80/20 . . . . .	64
7	Simulation 2: Percent Bias – ICC = 0.50; Missing = 30%; Split = 50/50 . . . . .	65
8	Simulation 2: Percent Bias – ICC = 0.50; Missing = 30%; Split = 80/20 . . . . .	66
9	Simulation 2: Coverage – ICC = 0.10; Missing = 30%; Split = 50/50 . . . . .	67
10	Simulation 2: Coverage – ICC = 0.10; Missing = 30%; Split = 80/20 . . . . .	68
11	Simulation 2: Coverage – ICC = 0.50; Missing = 30%; Split = 50/50 . . . . .	69
12	Simulation 2: Coverage – ICC = 0.50; Missing = 30%; Split = 80/20 . . . . .	70
13	Simulation 3: Percent Bias – ICC = 0.10; Missing = 30% . . . . .	71
14	Simulation 3: Percent Bias – ICC = 0.50; Missing = 30% . . . . .	72
15	Simulation 3: Coverage – ICC = 0.10; Missing = 30% . . . . .	73
16	Simulation 3: Coverage – ICC = 0.50; Missing = 30% . . . . .	74
17	Simulation 4: Percent Bias – ICC = 0.10; Missing = 30% . . . . .	75
18	Simulation 4: Percent Bias – ICC = 0.50; Missing = 30% . . . . .	76
19	Simulation 4: Coverage – ICC = 0.10; Missing = 30% . . . . .	77
20	Simulation 4: Coverage – ICC = 0.50; Missing = 30% . . . . .	78
21	Simulation 5: Percent Bias – ICC = 0.10; Missing = 30% . . . . .	79
22	Simulation 5: Percent Bias – ICC = 0.50; Missing = 30% . . . . .	80

23	Scatterplot of twelve clusters for $Y$ regressed on $X$ for $\chi^2(1)$ level-1 residual . .	81
24	Follow-up Simulation for Simulation 5 . . . . .	82
25	Scatterplot of twelve clusters for $Y$ regressed on $X$ for $\chi^2(15)$ level-1 residual . .	83

## LIST OF TABLES

2.1	Demonstrative Simulation . . . . .	19
4.1	Percent bias of posterior means for varying degrees of freedom. . . . .	50

## ACKNOWLEDGMENTS

This work was supported by Institute of Educational Sciences award R305D150056.

## VITA

- 2013            B.S. (Psychology), Arizona State University.
- 2015            M.A. (Psychology), Arizona State University.
- 2018            Candidate of Philosophy (Psychology), UCLA.
- 2015–present    Lead Programmer, Blimp Multilevel Imputation Software.

## PUBLICATIONS

**Keller, B. T.** & Enders, C. K. (2019). Blimp software manual (Version 2.0). Los Angeles, CA. Available at <http://www.appliedmissingdata.com/multilevel-imputation.html>

**Keller, B. T.**, & Du, H. (2019). A Fully Conditional Specification Approach to Multilevel Multiple Imputation with Latent Cluster Means. *Multivariate Behavioral Research*, 54(1), 149.

Enders, C. K., **Keller, B. T.**, & Levy, R. (2018). A fully conditional specification approach to multilevel imputation of categorical and continuous variables. *Psychological Methods*, 23(2), 298-317.

Enders, C. K., Du, H., & **Keller, B. T.** (in press). A Model-Based Imputation Procedure for Multilevel Regression Models with Random Coefficients, Interaction Effects, and Non-linear Terms. *Psychological Methods*.

Enders, C. K., Mistler, S. A., & **Keller, B. T.** (2016). Multilevel Multiple Imputation: A

Review and Evaluation of Joint Modeling and Chained Equations Imputation. *Psychological Methods*, 21(2), 222-240.

Geiser, C., Litson, K., Bishop, J., **Keller, B. T.**, Burns, G. L., Servera, M., & Shiffman, S. (2015). Analyzing person, situation and person  $\times$  situation interaction effects: Latent state-trait models for the combination of random and fixed situations. *Psychological Methods*, 20(2), 165-192.

Geiser, C., **Keller, B. T.**, & Lockhart, G. (2013). First-versus second-order latent growth curve models: some insights from latent state-trait theory. *Structural Equation Modeling: A Multidisciplinary Journal*, 20(3), 479-503.

Geiser, C., **Keller, B. T.**, Lockhart, G., Eid, M., Cole, D. A., & Koch, T. (2015). Distinguishing state variability from trait change in longitudinal data: The role of measurement (non) invariance in latent state-trait analyses. *Behavior Research Methods*, 47(1), 172-203.

# CHAPTER 1

## Introduction

Over the last few decades, a large body of research supports the use of multiple imputation as an appropriate method for handling missing data. The multiple imputation procedure was proposed by Rubin (2004) and has evolved over the years. In comparison to deletion methods (e.g., listwise deletion), multiple imputation has the ability to handle missing data under a less strict assumption about how the missing data arose. Despite the procedure first being published thirty years ago, only within the last decade have methodologists investigated imputation’s ability to handle interactions and other nonlinear terms. Methodologists originally suggested to impute the product terms separately, as if it was just another variable; however, this approach requires the same missing data assumption that the older deletion methods make (Enders, Baraldi, & Cham, 2014; Seaman, Bartlett, & White, 2012). More recently, advances have been made in the handling of nonlinear terms under a missing at random assumption (Bartlett, Seaman, White, & Carpenter, 2012, 2014; Goldstein, Carpenter, & Browne, 2014). These so-called “substantive model compatible”, henceforth referred to as model-based imputation, methods are closely related to the full Bayesian approach originally described by Ibrahim, Chen, and Lipsitz (2002) and subsequently investigated by Erler et al. (2016) and Zhang and Wang (2017) in the context of incomplete interactions. The purpose of this dissertation is to investigate the extension of model-based imputation to models of multilevel cross-interaction effects with missing data.

The organization of this chapter is as follows. First, I begin by offering a brief introduction to missing data theory. Second, I give a broad overview of the multiple imputation procedure.

Third, I give a brief introduction to Bayesian estimation of a two-level model with complete data. Fourth, I describe in detail the multilevel fully conditional specification (FCS) framework for multiple imputation.

## 1.1 Missing Data Mechanisms

The purpose of this section is to introduce some common concepts in the missing data literature. The two fundamental and distinct concepts that are often described are missing data patterns and missing data mechanisms. A missing data pattern describes the location of the missing observations in the data set and makes no attempt to describe how the observed pattern came to be. Conversely, a missing data mechanism provides a probabilistic account for how the observed missingness arose.

To concretely demonstrate missing data mechanisms, I first introduce some general notation for this section. Let  $Y = \{Y_1, \dots, Y_n\}$  be a vector of  $n$  random variables with a probability density function  $f(\cdot; \theta)$ , where the analytic goal is to make inferences about  $\theta$  (i.e., the parameters of the density). In addition to  $Y$ , there is also a vector of random binary indicator variables,  $M = \{M_1, \dots, M_n\}$ , with a probability mass function  $g(\cdot, \phi)$ , where  $\phi$  are nuisance parameters; thus, the observed missing data mechanism can be written as  $g(m | y, \phi)$ , which can be read as the probability of  $M$  taking on the value  $m$  given  $Y$  takes on the value of  $y$ . Therefore,  $m$  would be the observed missing data pattern and  $g(m | y, \phi)$  can be thought of as the missing data mechanism. Additionally, the vector  $y$  can be split into a missing vector and an observed vector:  $y = \{y^{(mis)}, y^{(obs)}\}$ , where the superscript ‘*(mis)*’ denotes the observations that are missing and ‘*(obs)*’ denotes the observations that are observed.

Little and Rubin (2002) defined three classifications of missing data mechanisms: missing completely at random, missing at random, and not missing at random. Missing completely at random (MCAR) can be thought of as data missing in a haphazard fashion, or “truly

random” in a colloquial sense. In terms of the probability mass function, under MCAR  $g(m | y, \phi)$  simplifies as follows.

$$g(m | y, \phi) = g(m, \phi) \tag{1.1}$$

Equation 1.1 can be read as the probability of missingness for  $Y$  is independent of the values of  $y$ , observed or missing. MCAR is a special case of the second missing data mechanism, missing at random (MAR). MAR states that the probability of missingness for  $Y$  is independent of the values of  $y^{(mis)}$ , but not  $y^{(obs)}$ . More formally written as follows.

$$g(m | y, \phi) = g(m | y^{(obs)}, \phi) \tag{1.2}$$

Rubin (1976) showed that MAR is a necessary assumption for making statistical inferences about  $\theta$  by ignoring the specific cause of the missing data (i.e., without directly modeling  $g(\cdot, \phi)$  in the analysis); thus, most current techniques for missing data assume MAR. Moving forward, I will always assume that the data are missing in such a way that MAR can be satisfied.

Finally, the third mechanism, not missing at random (NMAR), states that the probability of missingness for  $Y$  is dependent on the missing values of  $y$ .

$$g(m | y, \phi) = g(m | y^{(mis)}, \phi) \tag{1.3}$$

Current techniques to handle NMAR data require directly modeling  $g(m | y^{(mis)}, \phi)$ . These techniques are sensitive to misspecifications in the selection mechanism and will not be the focused of this dissertation.

## 1.2 Introduction to Multiple Imputation

Multiple imputation (MI) takes a Bayesian-based approach towards handling missing data (Rubin, 2004; Schafer, 1997) and consists of three major steps: An imputation step, an analysis step, and a pooling step. In the imputation step, the researcher “fills-in” or imputes the missing data with plausible values to obtain an imputed data set (i.e., data set with

filled-in values). The researcher repeats this process to obtain multiple imputed data sets with different plausible values. The imputation process is often achieved by using a Markov chain Monte Carlo (MCMC) method, such as a Gibbs sampler, to sample from a distribution of plausible values determined by the imputation model. Next, the analysis step is where the researcher analyzes these saved copies of data with an analysis model. Finally, the pooling step is when the researcher pools the estimates from the analysis by averaging estimates in accordance to specific formulas (see Rubin, 2004; Schafer, 1997). The motivation behind MI is to treat the missing data as a source of random variability that needs to be averaged over. While I have provided a very broad overview of the general steps of MI, the major focus of MI literature is on the imputation step and developing appropriate imputation models; thus, this is also the focus of the dissertation. In the subsequent sections, I will give a more detailed account of the imputation step and developing appropriate imputation models in the context of interactions and other nonlinear terms.

Currently, there are two major frameworks for specifying an imputation model: the joint model approach and fully conditional specification (also known as chained equations). The joint model approach (JM) uses a multivariate model to impute all missing variables simultaneously. For example, with single-level data JM can use a multivariate regression model with complete variables serving as predictors of the incomplete variables. In contrast to JM, fully conditional specification (FCS) uses a univariate model with a missing variable regressed on complete and previously imputed variables; therefore, FCS approximates the joint distribution with several univariate conditional distributions. Both frameworks have been extended to multilevel models, with most work focused on two-level data structures.

### **1.3 Bayesian Estimation of Two-level Model**

To facilitate my later discussion of multilevel imputation in the FCS framework, I will briefly describe Bayesian estimation of a two-level model. This section's focus is more of a high-level overview of the estimation process; therefore, I will largely omit the specific computational details. To illustrate a Bayesian estimation process for a two-level model, consider the fol-

lowing random intercept model with a normally distributed outcome

$$y_{ij} = \beta_0 + \beta_1 x_{ij} + \beta_2 z_j + u_{0j} + e_{ij} \quad (1.4)$$

where  $y_{ij}$  is the outcome measure for observation  $i$  in level-2 cluster  $j$ ,  $x_{ij}$  is a level-1 predictor with a slope  $\beta_1$ ,  $z_j$  is a level-2 predictor with a slope  $\beta_2$ , and  $\beta_0$  is the intercept. Furthermore, there is a level-1 residual,  $e_{ij}$ , that accounts for the deviation within a cluster, and there is a level-2 residual,  $u_{0j}$ , that accounts for the deviation between clusters. In line with multilevel models that are widely used in practice, I assume both residuals are normally distributed (i.e.,  $e_{ij} \sim \mathcal{N}(0, \sigma_e^2)$  and  $u_{0j} \sim \mathcal{N}(0, \sigma_{u_0}^2)$ ). I use this simple model to describe the procedure, but the basic estimation steps readily extend to even more complex models (e.g., inclusion of a random slope).

For my discussion of complete-data Bayes estimation, I will narrow my focus towards the conditional distributions used in a Gibbs sampler. In the case of two-level data, I must determine the conditional distributions of the regression parameters (i.e., so-called ‘fixed-effects’), the level-2 residuals (i.e., so-called ‘random-effects’), and finally the two variance parameters. For one iteration of the Gibbs sampler I first sample from the conditional distribution for the regression parameters, treating the other parameters and residuals as known. Next, the algorithm treats those regression parameters as known and then samples the level-2 residuals. The algorithm continues to sample each unknown conditional on the data, the priors, and the previously sampled parameters and residuals. Once all unknowns are sampled, the algorithm starts over again. These conditional distributions are available in a variety of resources (e.g., Browne, 1998).

More formally, the complete-data sampling steps for iteration  $t$  of the Gibbs sampler are as follows

$$\begin{aligned} \boldsymbol{\beta}^{(t)} &\sim MVN(\boldsymbol{\beta} \mid \mathbf{u}^{(t-1)}, \sigma_e^{2(t-1)}, \sigma_{u_0}^{2(t-1)}, \text{data}) \\ \mathbf{u}^{(t)} &\sim MVN(\mathbf{u} \mid \boldsymbol{\beta}^{(t)}, \sigma_e^{2(t-1)}, \sigma_{u_0}^{2(t-1)}, \text{data}) \\ \sigma_e^{2(t)} &\sim IG(\sigma_e^2 \mid \boldsymbol{\beta}^{(t)}, \mathbf{u}^{(t)}, \sigma_{u_0}^{2(t-1)}, \text{data}) \\ \sigma_{u_0}^{2(t)} &\sim IW(\sigma_e^2 \mid \boldsymbol{\beta}^{(t)}, \mathbf{u}^{(t)}, \sigma_e^{2(t)}, \text{data}) \end{aligned} \quad (1.5)$$

where  $MVN(\cdot)$  denotes a multivariate normal distribution,  $IG(\cdot)$  denotes an inverse gamma distribution, and  $IW(\cdot)$  denotes an inverse Wishart distribution. Equation 1.5 illustrates that each step conditions on the previous steps, updating the parameter or residual with the new sampled value. By running thousands of iterations of the Gibbs sampler characterized by Equation 1.5 and saving the parameter draws, I can obtain empirical estimates of the conditional distributions of the parameters and residuals. I can then use summary statistics (e.g., mean, median, standard deviation, etc.) to characterize the nature of these and obtain results that are analogous to a frequentist point estimate and standard errors. As I will discuss in the next section, FCS imputation uses the same estimation steps described above with an additional step that also updates the missing values.

## 1.4 FCS Imputation of Level-1 Incomplete Variable

FCS implements a separate imputation model for each incomplete variable, treating an incomplete variable as an outcome predicted by complete and previously imputed variables. For clustered data, the imputation model is often a univariate two-level regression model. To facilitate my discussion, I will consider the analysis model in Equation 1.4, assuming both  $Y$  and  $X$  are incomplete. FCS imputation applies the Bayesian estimation steps from Equation 1.5 with  $Y$  as the outcome, then proceeds to use the parameter values and residuals to draw imputations. Once  $Y$  is imputed, the algorithm then applies a separate set of Bayesian estimation steps with  $X$  as the outcome. As with  $Y$ ,  $X$  is imputed using the drawn parameter values and residuals. More concretely, the algorithm proceeds as follows

$$\begin{aligned}
 \theta_Y^{(t)} &\sim P(\theta_Y \mid y^{(t-1)}, x^{(t-1)}, z) \\
 Y^{(t)} &\sim P(Y \mid \theta_Y^{(t)}, x^{(t-1)}, z) \\
 \theta_X^{(t)} &\sim P(\theta_X \mid y^{(t)}, x^{(t-1)}, z) \\
 X^{(t)} &\sim P(X \mid \theta_X^{(t)}, y^{(t)}, z)
 \end{aligned} \tag{1.6}$$

where  $\theta_Y = \left\{ \boldsymbol{\beta}_{(Y)}, \mathbf{u}_{(Y)}, \sigma_{e_{(Y)}}^2, \sigma_{u_{(Y)}}^2 \right\}$  is a set of parameters and residuals for the regression of  $Y$  and I use  $P(\cdot)$  to denote a general probability distribution. The ‘(Y)’ subscript de-

notes that the parameter or residual is specific to the imputation model for  $Y$ . Additionally,  $\theta_X = \left\{ \boldsymbol{\beta}_{(X)}, \mathbf{u}_{(X)}, \sigma_{e(X)}^2, \sigma_{u(X)}^2 \right\}$  is a set of parameters and residuals for  $X$ 's imputation model. An understated point is that van Buuren's (Van Buuren, 2007, 2012; Van Buuren, Brand, Groothuis-Oudshoorn, & Rubin, 2006) original formulation of FCS uses the observed cases on the outcome variable (i.e., the target of imputation) to draw the necessary parameters and cluster-level residuals (e.g., for  $\theta_Y$  the algorithm would use only the observed cases for  $Y$ ). My notation is in line with the standard Gibbs sampler that uses both the observed and missing parts of the data. Using the observed cases tends to converge faster but can lead to computational problems when a cluster has a high missing data rate.

After applying the Bayesian estimation sequence to sample the necessary parameters and residuals, an additional step is required for imputation of the missing values. The imputation of  $Y$  is based on a two-level model that features  $X$  (imputed at the previous iteration) and the complete covariate as predictors. The imputations are drawn from a normal distribution centered at the predicted score of the imputation model with a spread defined by the level-1 residual variance. More formally, for iteration  $t$  the imputation of the  $i^{\text{th}}$  observation within cluster  $j$  is as follows

$$y_{ij}^{(t)} \sim \mathcal{N}(\beta_{0(Y)}^{(t)} + \beta_{1(Y)}^{(t)} x_{ij}^{(t-1)} + \beta_{2(Y)}^{(t)} z_j + \beta_{3(Y)}^{(t)} \bar{x}_j^{(t-1)} + u_{0_j(Y)}^{(t)}, \sigma_{e(Y)}^{2(t)}) \quad (1.7)$$

where ' $(t-1)$ ' superscript on  $X$  and  $Z$  denotes that they were filled-in at iteration  $t-1$  if missing, otherwise they are the observed values. Finally, following Snijders and Bosker (2012) notation, I use a single 'bar' above a variable (i.e.,  $\bar{x}$ ) to denote a cluster mean at the level-2 sampling unit. Despite not being present in the analysis model (Equation 1.4), I include the cluster means in the imputation model. The literature suggests that the cluster means are necessary to allow within and between-cluster covariance matrices to differ (Carpenter & Kenward, 2013; Keller, Enders, & Kim, 2019; Mistler & Enders, 2017). Likewise, the imputation of  $X$  is based on a two-level model that features the previously-imputed  $Y$  and the complete covariate as predictors.

$$x_{ij}^{(t)} \sim \mathcal{N}(\beta_{0(X)}^{(t)} + \beta_{1(X)}^{(t)} y_{ij}^{(t)} + \beta_{2(X)}^{(t)} z_j + \beta_{3(X)}^{(t)} \bar{y}_j^{(t)} + u_{0_j(X)}^{(t)}, \sigma_{e(X)}^{2(t)}) \quad (1.8)$$

Again, the ‘(X)’ subscript denotes that the parameter or residual is specific to variable  $X$  and these parameters and residual terms are sampled when  $\theta_X$  is drawn in Equation 1.6. The steps for drawing  $\theta_X$  are a straightforward application of Equation 1.4, but with  $X$  as the outcome.

The algorithm continues to iterate over the process described by Equation 1.6 for thousands of iterations. Once the Gibbs sampler reaches a stationary distribution, I save initial imputations and proceed to save additional imputations after a thinning interval determined by convergence diagnostics (e.g., the potential scale reduction statistic; Gelman & Rubin, 1992). After all imputations are saved, the imputed data sets are analyzed and the parameter estimates and standard errors are aggregated (Little & Rubin, 2002; Rubin, 2004; Van Buuren, 2012). Moving forward, the subsequent sections will outline the inclusion of incomplete level-2 variables.

## 1.5 FCS Imputation of Level-2 Incomplete Variable

Often, researchers are interested in variables measured at each level of the data hierarchy. To construct an appropriate imputation model, the model must preserve the associations among variables measured at each level; therefore, a variable at one level must be included in the imputation models at all other levels. In the two-level context, the literature suggests the use of cluster means as a solution to preserve the cross-level associations (Carpenter & Kenward, 2013; Gelman & Hill, 2006; Yucel, 2008). An alternative method is to use latent cluster means. In a two-level context, Grund, Lüdtke, and Robitzsch (2016) investigated the use of latent cluster means and found that manifest cluster means performed similarly to latent cluster means unless the data are extremely unbalanced (i.e., the within-cluster size greatly varies from cluster to cluster). In models where it is of interest to model heterogeneous within-cluster variances (Raudenbush & Bryk, 2002, pp. 130-133), it may also be useful to include other score summaries (e.g., within-cluster variance estimates) in the level-2 imputation model, but I restrict my focus to homogeneous models that do not require these alternate

specifications. To facilitate my discussion of incomplete level-2 variables, suppose that  $Z$  is now an incomplete level-2 variable.

To impute the level-2 variable,  $Z$ , the algorithm must aggregate the two-level data set into a single-level data set. To illustrate, consider a two-level data set with repeated measures nested within participants. Level-2 imputation requires a single-level data matrix aggregated across participants; thus, if there are  $J$  total participants, then the aggregated single-level data matrix would contain  $J$  rows. By averaging  $Y$  and  $X$  within each level-2 cluster, I now have a single-level structure with the cluster-mean of  $Y$  and  $X$  at each observation. As a result, the columns for the single-level data matrix correspond to  $\bar{Y}$ ,  $\bar{X}$ , and  $Z$ .

With the aggregated single-level data matrix, I construct a single-level regression model where the level-2 variable  $Z$  is the criterion that is regressed on the cluster means of the previously imputed and complete variables. Therefore, the imputation model for the  $j^{th}$  observation of  $Z$  at iteration  $t$  is as follows.

$$z_j^{(t)} \sim \mathcal{N}(\beta_{0(Z)}^{(t)} + \beta_{1(Z)}^{(t)} \bar{y}_j^{(t)} + \beta_{2(Z)}^{(t)} \bar{x}_j^{(t)}, \sigma_{e(Z)}^2) \quad (1.9)$$

The ‘(Z)’ subscript denotes that the parameter or residual is specific to the variable  $Z$ . The imputations described by Equation 1.9 are obtained by drawing from a distribution centered at the predicted score of a single-level regression model. The algorithm draws the necessary parameters and residuals for the single-level regression model based on the conditional distributions used in a Bayesian estimation (e.g., Gelman et al., 2013; Lynch, 2007). If there were more incomplete level-2 variables, then the algorithm would continue to impute the additional level-2 variables one at a time. Like the procedure at level-1, the next incomplete level-2 variable would become an outcome in a model that has the same structure as that in Equation 1.9, with  $Z^{(t)}$  (i.e., the imputations or observed values at iteration  $t$ ) serving as a predictor. This process would continue until all incomplete level-2 variables are imputed.

## CHAPTER 2

### Literature Review

#### 2.1 Imputation of Single-level Interaction Effects

Thus far, I have reviewed the algorithmic work required to implement two-level FCS imputation. Moving forward, I will change my focus to imputing interactions. Because virtually all work has been conducted in the single-level context, my review will be based on single-level models; however, this work readily extends to two-levels as well. Schafer (1997) first identified that imputation models at the time were not readily able to handle interactions and other nonlinear effects (e.g., higher order polynomials). Allison (2002) later proposed three different methods (simply labeled Method 1, Method 2, and Method 3) in an attempt to handle nonlinear effects; however, Allison (2002) did not investigate if they were biased. Von Hippel (2009) was the first methodological work that investigated the original methods (and possible variants of them) that Allison (2002) proposed; therefore, my discussion of these three methods will largely follow that of von Hippel (2009). The three methods that von Hippel (2009) discussed are: Stratify-then-impute (Allison’s Method 2), impute-then-transform (Allison’s Method 1), and transform-then-impute (Allison’s Method 3).

To facilitate my discussion of imputation when interactions are present, suppose the substantive model of interest is a moderated regression with three variables,  $X$ ,  $Y$ , and  $Z$ . In this example,  $Y$  will be the criterion regressed on  $X$ ,  $Z$ , and the product of  $X$  and  $Z$ ; thus, the regression equation for the  $i^{th}$  individual is as follows.

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 x_i z_i + e_i \tag{2.1}$$

For the remainder of my dissertation I will reserve ‘ $\beta$ ’ to denote a regression parameter for the

substantive model and ‘ $e$ ’ to denote a residual for the substantive model. For the subsequent discussion, I will also assume that only  $X$  is incomplete following a MAR mechanism.

Stratify-then-impute (sometimes referred to as separate group models), requires one of the variables in the interaction term to be complete and discrete. Stratify-then-impute is analogous to a multiple group analysis, estimating different imputation models for each level of the categorical predictor (Enders & Gottschall, 2011). To illustrate, I specify a binary  $Z$  (i.e., coded 0 and 1) for my discussion of stratify-then-impute. Although generally an algorithm uses all data to estimate the imputation model, stratify-then-impute partitions the data set by the grouping variable  $Z$ ; thus, there are two separate imputation models for the incomplete variable,  $X$  (i.e. an imputation model for the  $j^{th}$  individual in the group where  $Z$  is equal to 0 and an imputation model for the  $k^{th}$  individual in the group where  $Z$  is equal to 1).

$$\begin{aligned} x_j^{(t)} &\sim \mathcal{N}(\alpha_0 + \alpha_1 y_j, \sigma_r^2(Z=0)) \\ x_k^{(t)} &\sim \mathcal{N}(\gamma_0 + \gamma_1 y_k, \sigma_r^2(Z=1)) \end{aligned} \tag{2.2}$$

Note for simplicity, I have excluded the iteration superscripts on the parameters. The important feature of Equation 2.2 is the fact that the parameters in the model are different in each group (denoted with different Greek letters or subscripted  $Z = 0$  and  $Z = 1$ ). Additionally, I exclude  $Z$  from the imputation model because the group differences are inherently modeled by the use of separate parameters. To implement this method in practice, a researcher splits the data set by the grouping variable and then runs an imputation procedure on each new data set. One limiting factor of stratify-then-impute is when one level of the category has very few observations; otherwise, simulation work suggests the method performs well (Enders & Gottschall, 2011).

The second method discussed by von Hippel (2009) is impute-then-transform. Impute-then-transform and its variants rely on first imputing the incomplete predictor and then computing the product term based upon the imputation. Returning to the example in Equation 2.1, impute-then-transform would first perform imputation of the variable  $X$ . Then, after imputation, the procedure computes the product of  $X$  and  $Z$  for when  $X$  is observed

and the product of the imputation and  $Z$  when  $X$  is missing. Impute-then-transform uses the following imputation model for the  $i^{\text{th}}$  observation of  $X$ .

$$x_i^{(t)} \sim \mathcal{N}(\alpha_0 + \alpha_1 y_i + \alpha_2 z_i, \sigma_r^2) \quad (2.3)$$

Equation 2.3 is based upon Allison’s (2002) initial recommendation for Method 1. Von Hippel (2009) provides analytic work to show that using the imputation model in Equation 2.3 inflated the residual variance and attenuated the slope parameters in linear regressions. An alternative specification that is sometimes used is to include an interaction in the imputation model between  $Y$  and  $Z$ .

$$x_i^{(t)} \sim \mathcal{N}(\alpha_0 + \alpha_1 y_i + \alpha_2 z_i + \alpha_3 y_i z_i, \sigma_r^2) \quad (2.4)$$

The problem with both these imputations models (i.e., Equation 2.3 and 2.4) is that they do not accurately correspond to the “true” conditional distribution for  $X$  given  $Y$  and  $Z$ ; thus, the imputation model of  $X$  is not correctly specified and the imputed values will not be consistent with the nonlinear nature between  $X$  and  $Y$ . A third option available when  $Y$  is also missing, known as passive imputation (Royston, 2005), uses an imputation model analogous to Equation 2.4 (or sometimes Equation 2.3) for the imputation of  $X$ . For the imputation of  $Y$ , the imputation model is of same form as the analysis model in Equation 2.1. The key difference is that the product terms are computed within each iteration (e.g., at iteration  $t$  for observation  $i$ , using  $x_i^{(t)}$  to compute the interaction in the imputation of  $y_i^{(t)}$ ). Simulations and analytic work have illustrated that passive imputation still produces biased imputations under both MCAR and MAR mechanisms (Seaman et al., 2012; von Hippel, 2009). This bias can be attributed to passive imputation still not accurately specifying the correct conditional distribution for  $X$ .

The final imputation strategy discussed by von Hippel (2009) is known as transform-then-impute (sometimes referred to as “just another variable” approach, or JAV; White, Royston, & Wood, 2011). Transform-then-impute or JAV creates a new variable,  $W$ , that represents the product of  $X$  and  $Z$ . This product variable is missing when  $X$  is missing and is complete when  $X$  is complete. The new variable is treated as if it was just another variable; therefore,

an imputation model is specified for  $W$  because it has incomplete observations. Returning to the moderate regression example, the FCS imputation models for  $X$  and  $W$  are as follows.

$$\begin{aligned} x_i^{(t)} &\sim \mathcal{N}(\alpha_0 + \alpha_1 y_i + \alpha_2 z_i + \alpha_3 w_i^{(t-1)}, \sigma_{r(X)}^2) \\ w_i^{(t)} &\sim \mathcal{N}(\gamma_0 + \gamma_1 y_i + \gamma_2 z_i + \gamma_3 x_i^{(t)}, \sigma_{r(W)}^2) \end{aligned} \tag{2.5}$$

Traditionally,  $W$  is then used in place of the product of  $X$  and  $Z$  in the subsequent analysis step. In the case that  $X$  and  $Z$  are used instead of  $W$ , von Hippel (2009) found that this did no better than impute-then-transform. Von Hippel (2009) determined that transform-then-impute was the best method of the ones he investigated; however, von Hippel’s study was severely limited by only investigating data that followed an MCAR mechanism. While transform-then-impute became the main method for handling interactions in imputation, later simulation work showed that the method produced bias parameter estimates under the MAR mechanism (Enders et al., 2014; Seaman et al., 2012; Zhang & Wang, 2017).

To better understand the problem with transform-then-impute, I return to the imputation model of  $W$  in Equation 2.5. Importantly, this conditional distribution is misspecified because  $W$  is a product of two normally distributed variables and therefore cannot be normally distributed. Seaman et al. (2012) used Kullback-Leibler divergence to explain why transform-then-impute still works under the MCAR assumption. Conceptually, Kullback-Leibler divergence is a measure of how an arbitrary distribution differs from another distribution. They illustrated that the population parameters minimized the Kullback-Leibler divergence between the misspecified conditional distribution of  $W$  in Equation 2.5 and the product’s true distribution. These population parameters are consistently estimated under the MCAR mechanism, leading to consistent parameter estimates. In contrast, under an MAR mechanism the population parameters cannot be consistently estimated and will lead to inconsistent estimates in the analysis model. Additionally, Seaman et al. (2012) raised the point that even under an MCAR mechanism the estimates of the sampling variance may still be biased. Their rationale behind this belief is that the derivations of the pooling rules of sampling variance given by Rubin (2004) assumes that the imputation model is correctly specified.

## 2.2 Model-Based Imputation

So far, I have discussed the idea of misspecified imputation models leading to biases in statistical estimates. When nonlinearity is present, all methods except stratify-then-impute incorrectly specify the conditional distribution of a missing covariate; therefore, the solution is to correctly derive conditional imputation models that accurately reflect the true distribution. Bartlett, Seaman, White, and Carpenter (2014) provided derivations for a general imputation model that ensures that the conditional distribution of a covariate is correctly specified for a given substantive model. Their derivations are closely related to a full Bayesian approach for handling missing data originally described by Ibrahim, Chen, and Lipsitz (2002). The full Bayesian approach has also recently been investigated in the context of interactions by Erler et al. (2016) and Zhang and Wang (2017). The main focus of my dissertation is to investigate the application of Bartlett et al.'s (2014) approach to cross-level interactions; thus, I will first focus on their work in the single-level context and thereon its extension to two-levels (Enders, Du, & Keller, in press).

I return to the moderated regression example, treating Equation 2.1 as the correctly specified substantive model. Recall from the example that  $X$  is missing; therefore, I must derive a correctly specified imputation model for  $X$ . In order to correctly specify an imputation model for  $X$ , I must sample from the true conditional probability of  $X$  given  $Y$  and  $Z$ ,  $P(X | Y, Z)$ . Bartlett et al. (2014) re-expressed this conditional probability as follows.

$$\begin{aligned} P(X | Y, Z) &= \frac{P(Y, X, Z)}{P(Y, Z)} \\ &= \frac{P(Y | X, Z)P(X | Z)}{P(Y | Z)} \\ &\propto P(Y | X, Z)P(X | Z) \end{aligned} \tag{2.6}$$

Equation 2.6 illustrates that  $P(X | Y, Z)$  is proportional to the product of two probabilities. The first conditional probability,  $P(Y | X, Z)$ , is the conditional probability of  $Y$  given  $X$  and  $Z$ , which corresponds to the substantive model. For reasons that become apparent later,

I rewrite the substantive model in Equation 2.1 in terms of its probability density function for the  $i^{\text{th}}$  observation.

$$f(y_i | x_i, z_i, \theta) = \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp \left[ -\frac{y_i - (\beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 x_i z_i)}{2\sigma_e^2} \right] \quad (2.7)$$

I use  $\theta$  denote to the set of parameters for substantive model;<sup>1</sup> therefore, for the moderated regression example  $\theta = \{\beta_0, \beta_1, \beta_2, \beta_3, \sigma_e^2\}$ . To reduce the redundancy in equations, moving forward I will write a normal probability density function as  $\varphi(x; \mu, \sigma^2)$ ; thus, Equation 2.6 would be expressed as  $\varphi(y_i; \beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 x_i z_i, \sigma_e^2)$ .

Turning to  $P(X | Z)$ , this conditional probability can be expressed as a ‘‘covariate model’’ for  $X$ . In the moderated regression example, I define this covariate model as the regression of  $X$  on  $Z$ ; thus, the model for the  $i^{\text{th}}$  observation is as follows.

$$x_i = \alpha_{0(X)} + \alpha_{1(X)} z_i + r_{i(X)} \quad (2.8)$$

For the covariate model, I use ‘ $\alpha$ ’ to denote a regression parameter, ‘ $r$ ’ to denote the residual, and subscript ‘ $(X)$ ’ to denote that the parameter or residual is specific for the variable  $X$ . Under the typical assumption of the residuals being normally distributed, I can write the probability density function for Equation 2.8 for the  $i^{\text{th}}$  observation as follows.

$$f(x_i | z_i, \phi_X) = \varphi(x_i; \alpha_{0(X)} + \alpha_{1(X)} z_i, \sigma_{r(X)}^2) \quad (2.9)$$

I let  $\phi_X = \{\alpha_{0(X)}, \alpha_{1(X)}, \sigma_{r(X)}^2\}$  be the set of parameters that characterize the covariate model for  $X$ .

Now that I have parameterized models for  $P(Y | X, Z)$  and  $P(X | Z)$  (i.e., Equations 2.7 and 2.9), the imputations for  $X$  are drawn from the conditional probability of  $X$  given  $Y$ ,  $Z$ , and the parameters,  $\{\theta, \phi_{(X)}\}$ . In contrast to traditional imputation methods, I ensure that the imputations are correctly specified with the substantive model by explicitly using the substantive model’s parameters to characterize  $X$ ’s conditional distribution. For iteration  $t$ ,

---

<sup>1</sup>Note that when I refer to ‘‘the set of parameters for substantive model,’’ I refer to the parameters in the model, not the values in the population. For example,  $\theta^{(t)}$  refers to values of those parameters drawn at iteration  $t$ , not the values at the population level.

the substantive model compatible imputation algorithm takes the following form to impute  $X$ .

$$\begin{aligned}\theta^{(t)} &\sim P(\theta \mid Y, X^{(t-1)}, Z) \\ \phi_X^{(t)} &\sim P(\phi_X \mid X^{(t-1)}, Z) \\ X^{(t)} &\sim P(X \mid Y, Z, \theta^{(t)}, \phi_X^{(t)})\end{aligned}\tag{2.10}$$

The algorithm starts an iteration by conditioning on the previously imputed and observed values of  $X$  in order to sample values for the parameters of the substantive model. Next, the algorithm proceeds to condition on the previously imputed and observed values of  $X$  to sample values for parameters of the covariate model. Finally, imputations are drawn by conditioning on the sampled values of the substantive model parameters, sampled values of the covariate model parameters, and the observed data. Following the derivation in Equation 2.6, for iteration  $t$ , I express up to proportionality the density of the imputation model for the  $i^{\text{th}}$  observation of  $X$  as a product of two densities.

$$f(x_i \mid y_i, z_i, \theta^{(t)}, \phi_X^{(t)}) \propto f(y_i \mid x_i, z_i, \theta^{(t)})f(x_i \mid z_i, \phi_X^{(t)})\tag{2.11}$$

Conceptually, the function in Equation 2.11 can be thought of as the likelihood of the substantive model acting as a weight that ensures that the imputations account for the nonlinearity (i.e., interaction) present in the substantive model. In contrast to transform-then-impute, the imputation model does not impose any distributional assumptions on the interaction of  $X$  and  $Z$ . Instead, the imputation model correctly specifies the conditional distribution of  $X$  in a fashion that is consistent with the nonlinearity (i.e., using the parametrization of the substantive model); therefore, the interaction term is never imputed. Instead, the interaction is computed when  $\theta^{(t)}$  is drawn. To sample from the function in Equation 2.11, Bartlett et al. (2014) outlined an implementation of rejection sampling (Ripley, 1987); however, one drawback of rejection sampling is, as the dimensionality of the problem increases, rejection sampling becomes computationally inefficient (i.e., it becomes more and more unlikely to accept a sample; Gilks, Best, & Tan, 1995; Murphy, 2012). Because of the high dimensional nature of multilevel models, a Metropolis sampling step is a more viable and efficient answer; therefore, I will proceed by outlining the Metropolis sampling step.

## Metropolis Sampling Step

In general, the Metropolis algorithm samples values from a desired target probability distribution and can be embedded within a Gibbs sampler (Gelman et al., 2013; Lynch, 2007). One important feature of the Metropolis algorithm is that the probability distribution only needs to be specified by a function that is proportional to the target density. In the moderated regression example, the proportional function is given in Equation 2.11. The algorithm proceeds by generating a trial value of the imputation and then accepting or rejecting the imputation with some probability. Upon rejection of the trial value, the previous iteration’s value is used. To obtain the trial value I use what is referred to as a “random-walk” Metropolis step. A random-walk Metropolis step works by using a jumping distribution to take a random deviation from the previous iteration’s value. For the moderated regression example, I will use a normal jumping distribution centered at the previous iteration’s imputation; thus, for iteration  $t$ , the trial value for the imputation of the  $i^{th}$  observation is generated from the following distribution.

$$x'_i \sim \mathcal{N}(x_i^{(t-1)}, c\sigma_{r(X)}^{2(t)}) \quad (2.12)$$

I use  $x'_i$  to denote the trial value of the imputation for  $x_i$ . The jumping distribution is centered at the imputed value of  $x_i$  at iteration  $t - 1$ . I use the residual variance for  $X$  at iteration  $t$  (i.e.,  $\sigma_{r(X)}^{2(t)}$ ) to scale the jumping distribution based on the observed data. I then multiply the residual variance by an arbitrary constant,  $c$ . I adaptively tune the scale of the distribution by increasing and decreasing this arbitrary constant to obtain an ideal acceptance rate of the trial value. The literature recommends an ideal acceptance rate to be between twenty-five to forty-five percent of the time (Gelman et al., 2013). Generally, it is recommended to tune this constant during the burn-in iterations until the acceptance rate lies within the ideal range.

To determine the probability of accepting the trial imputation for  $X$ , the algorithm calculates the importance ratio (sometimes referred to as acceptance ratio; denoted as  $IR$ ) and sets the probability of acceptance equal to  $\min[IR, 1]$ . To calculate the importance ratio,

I divide the proportional function evaluated at the trial value by the proportional density evaluated at the previously imputed value. Returning to the moderated regression example, the importance ratio is calculated at iteration  $t$  as follows.

$$IR = \frac{f(y_i | x'_i, z_i, \theta^{(t)})f(x'_i | z_i, \phi_X^{(t)})}{f(y_i | x_i^{(t-1)}, z_i, \theta^{(t)})f(x_i^{(t-1)} | z_i, \phi_X^{(t)})} \quad (2.13)$$

In Equation 2.13, the numerator is the product of the likelihoods given in Equations 2.7 and 2.9 evaluated at the trial imputation and the denominator is the product of the likelihoods given in Equations 2.7 and 2.9 evaluated at the previous iteration's imputation. Finally, with the importance ratio computed, the following decision rule is used to determine the value of  $x_i$  at iteration  $t$

$$x_i^{(t)} = \begin{cases} x'_i & \text{if } u = 1 \\ x_i^{(t-1)} & \text{if } u = 0 \end{cases} \quad (2.14)$$

$$u \sim B(1, \min[IR, 1])$$

where  $B(n, p)$  is a Binomial distribution. As denoted by the subscripting, each draw is done at the observation level, requiring a separate Metropolis step for each incomplete observation.

## Demonstrative Simulation

To illustrate the performance of an imputation model that is correctly specified with the substantive analysis model, I ran a small simulation. The simulation follows similar conditions to those presented in Bartlett et al. (2014) and mirrors exactly the moderated regression example presented thus far. The analysis model of interest is given in Equation 2.1, with  $X$  incomplete and the other variables complete. The correlation between  $X$  and  $Z$  was generated to equal 0.50 and the population parameters of the analysis model are given in the first column of Table 2.1. To simulate the MAR mechanism, I used a latent variable formulation of logistic regression (Agresti, 2012; Johnson & Albert, 2006) to define a latent propensity of missingness for  $X$  dependent on the values of  $Z$ ; therefore, the inclusion of  $Z$  satisfies the MAR assumption. Parameters for the missing data model were selected to

induce a 25% missing data rate on  $X$ . I used a sample size of 250 with 2000 replications in total. Finally, 25 imputations were used for each replication. I imputed the data using a substantive compatible model and the transform-then-impute method. Imputation for both methods were done via Blimp (Keller & Enders, 2019).

Table 2.1: Demonstrative Simulation

Parameter	True Value	Estimates				Percent Bias	
		Estimation Method					
		JAV	MBI	JAV	MBI	JAV	MBI
Intercept	5.00	5.08	5.00	1.63	-0.03		
$X$ slope	1.00	1.09	1.00	9.29	0.50		
$Z$ slope	1.00	1.22	1.00	22.10	-0.14		
$XZ$ slope	1.00	1.09	1.00	8.95	0.25		
Residual Var.	4.00	4.35	3.99	8.78	-0.35		

*Note:* JAV is the Transform-then-impute (“Just another variable”) method. MBI, was using a model-based imputation routine. Both methods were implemented within an FCS framework.

The results of the simulation are presented in Table 2.1. As mentioned above, the first column provides the true values of each parameter. The next two columns present the average of the pooled estimates across all 2000 replications (rounded to two decimal places). Finally, the last two columns present the percent bias (i.e.,  $(\text{estimate} - \text{true value}) / (\text{true value}) \times 100$ ) based on the unrounded estimates. As Table 2.1 illustrates, the transform-then-impute approach remained under the recommended 10% bias threshold (Kaplan, 1988) for all but one coefficient. The coefficient for  $Z$  has a substantial amount of bias at 22%. In comparison, the substantive model compatible imputation approach had virtually no bias. This simulation serves to illustrate that when the assumptions are met (e.g., a correctly specified analysis model and covariate distribution), the procedure performs well in comparison to transform-then-impute or JAV. This performance could vary with different configurations

of missingness or percentage of missing data. Nevertheless, the simulation illustrates the effectiveness of the procedure at estimating models with interaction effects.

## Multivariate Missing Data Problems

Thus far, I have only considered a univariate missing data problem. In the case of multiple incomplete variables, I could construct correctly specified imputation models in both a JM framework and an FCS framework. Bartlett et al. (2014) derived both approaches for single-level models. Additionally, Goldstein et al. (2014) discussed specifically the single-level JM approach and the usage of a Metropolis step to sample from the density. Because the main focus of my proposal is FCS, I will focus on the corresponding derivation for FCS given by Bartlett et al. (2014) and the extension to multilevel models (Enders et al., in press).

Returning to the moderated regression example given in Equation 2.1, I will now assume that both  $Y$  and  $Z$  are incomplete under an MAR mechanism. For the imputation of  $Y$ , I specify the imputation model to be equivalent to the substantive model. Another option would be to specify a model with the substantive model nested within it (i.e., to include auxiliary variables that might make the MAR assumption more tenable). Because the substantive model is always correctly specified with itself, no special derivation or corresponding Metropolis step is needed to draw imputations for  $Y$ . Therefore, I express the imputation model for iteration  $t$  as follows.

$$y_i^{(t)} \sim \mathcal{N}(\beta_0^{(t)} + \beta_1^{(t)}x_i^{(t-1)} + \beta_2^{(t)}z_i^{(t-1)} + \beta_3^{(t)}x_i^{(t-1)}z_i^{(t-1)}, \sigma_e^{2(t)}) \quad (2.15)$$

I include the ‘ $(t - 1)$ ’ superscript to indicate that  $X$  and  $Z$  are imputed at iteration  $t-1$  if missing; otherwise, the observed values are used. Moreover, I include the ‘ $(t)$ ’ superscript on the parameters to more explicitly indicate that the parameters in Equation 2.15 are sampled at iteration  $t$ . These parameters are sampled prior to imputation from their conditional distributions via a Gibbs sampler.

In contrast to  $Y$ , the imputation of  $Z$  follows the same logic as  $X$ . Using an analogous derivation in Equation 2.6, the conditional probability for  $Z$  can be expressed up to proportionality as follows.

$$P(Z | Y, X) \propto P(Y | X, Z)P(Z | X) \quad (2.16)$$

The conditional distribution given by  $P(Y | X, Z)$ , coincides with the density in Equation 2.7, and the conditional distribution for the covariate model,  $P(Z | X)$  reflects the regression of  $Z$  on  $X$ ; thus, the density for  $Z$ 's covariate model for observation  $i$  is as follows.

$$f(z_i | x_i, \phi_Z) = \varphi(z_i; \alpha_{0(Z)} + \alpha_{1(Z)}x_i, \sigma_{r(Z)}^2) \quad (2.17)$$

I define the set of parameters that characterize the covariate model for  $Z$  as

$\phi_Z = \{\alpha_{0(Z)}, \alpha_{1(Z)}, \sigma_{r(Z)}^2\}$ . Now that both  $Y$  and  $Z$  require imputations, I modify the algorithm given in Equation 2.10. For iteration  $t$ , the model-based procedure imputes  $Y$ ,  $X$ , and  $Z$  using the following algorithm.

$$\begin{aligned} \theta^{(t)} &\sim P(\theta | Y^{(t-1)}, X^{(t-1)}, Z^{(t-1)}) \\ Y^{(t)} &\sim P(Y | X^{(t-1)}, Z^{(t-1)}, \theta^{(t)}) \\ \phi_X^{(t)} &\sim P(\phi_X | X^{(t-1)}, Z^{(t-1)}) \\ X^{(t)} &\sim P(X | Y^{(t)}, Z^{(t-1)}, \theta^{(t)}, \phi_X^{(t)}) \\ \phi_Z^{(t)} &\sim P(\phi_Z | X^{(t)}, Z^{(t-1)}) \\ Z^{(t)} &\sim P(Z | Y^{(t)}, X^{(t)}, \theta^{(t)}, \phi_Z^{(t)}) \end{aligned} \quad (2.18)$$

First, model-based imputation samples the parameters for the imputation of the outcome. Second, model-based imputation uses the sampled parameters to impute any missing data on the outcome. Third, model-based imputation imputes the missing covariates one at a time, by sampling the parameters of a covariate model and then imputing the covariate via a Metropolis step. Once all covariates are imputed, the algorithm the next iteration begins, and the entire process starts over again.

## 2.3 Multilevel Model-Based Imputation

So far, I have presented model-based imputation exclusively for handling single-level models. Erler et al. (2016) investigated incomplete level-2 covariates with cross-level interaction using a full Bayesian modeling approach and gave accompanying JAGS (Plummer, 2016) syntax for their implementation. Enders et al. (in press) extended a model-based imputation approach up to three-levels with any configuration of missing data and provided an implementation of the procedure in the Blimp program (Keller & Enders, 2019). The full Bayesian and model-based imputation approaches are equivalent in specific configurations of missing variables and models, but I will focus solely on multilevel model-based imputation.

In addition to nonlinear terms (e.g., cross-level interactions), traditional FCS imputation models are incompatible when random slopes are present (Enders, Hayes, & Du, 2018; Keller et al., 2019); thus, I will specify an analysis model with both sources of incompatibility. I will use the following two-level analysis model.

$$\begin{aligned}
 y_{ij} &= \beta_0 + \beta_1 x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0j} + u_{1j} x_{ij} + e_{ij} \\
 \begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}_u = \begin{bmatrix} \sigma_{u_0}^2 & \sigma_{u_0 u_1} \\ \sigma_{u_0 u_1} & \sigma_{u_1}^2 \end{bmatrix} \right) \\
 e_{ij} &\sim \mathcal{N}(0, \sigma_e^2)
 \end{aligned} \tag{2.19}$$

In the analysis example,  $Y$  is the level-1 incomplete dependent variable,  $X$  is the level-1 incomplete predictor,  $Z$  is the level-2 incomplete predictor, and there is a cross-level product term between  $X$  and  $Z$ . Each predictor has a corresponding regression coefficient (denoted with a  $\beta$  and a subscript), and the intercept is given as  $\beta_0$ . Finally, the level-2 residuals (i.e.,  $u_0$  and  $u_1$ ) have a multivariate normal distribution with means of zero and an unstructured covariance matrix.

## Imputing Level-1 Variables

Conceptually, the model-based imputation algorithm for the moderated two-level example does not change from the single-level description in Equation 2.18. Rather, I need only change the specifications of the conditional distributions to be in line with the multilevel nature of the two-level example. For the substantive model, I now use Equation 2.19, and  $\theta$  now includes the multilevel residuals (i.e.,  $u_0$  and  $u_1$ ) and their accompanying parameters (i.e.,  $\Sigma_u$ ). The drawing of these parameters and residuals is analogous to the draws described in Equation 1.5 but with the modification of drawing a level-2 covariance matrix instead of a single between-cluster variance. For the imputation of  $Y$  at iteration  $t$ , I draw imputations as follows.

$$y_{ij}^{(t)} \sim \mathcal{N}(\beta_0^{(t)} + \beta_1^{(t)}x_{ij}^{(t-1)} + \beta_2^{(t)}z_j^{(t-1)} + \beta_3^{(t)}x_{ij}^{(t-1)}z_j^{(t-1)} + u_{0j}^{(t)} + u_{1j}^{(t)}x_{ij}^{(t-1)}, \sigma_e^{2(t)}) \quad (2.20)$$

Similar to the previous section, the ‘ $(t)$ ’ superscript on a parameter or residual represents that the value was sampled at iteration  $t$ . Just like in the single-level context (e.g., Equation 2.15), in the multilevel context the imputation model contains the interaction. This ensures that the imputation model is correctly specified.

Turning to the imputation of the level-1 covariate  $X$ , the derivations do not change from Equation 2.6, but the models employed and the resulting densities do. The covariate model for  $X$  is now a multilevel random intercept regression model with  $X$  regressed on all other covariates; therefore, the model is as follows.

$$\begin{aligned} x_{ij} &= \alpha_{0(X)} + \alpha_{1(X)}z_j + d_{0j(X)} + r_{ij(X)} \\ d_{0j(X)} &\sim \mathcal{N}(0, \sigma_{d_{0j(X)}}^2) \\ r_{ij(X)} &\sim \mathcal{N}(0, \sigma_{r_{ij(X)}}^2) \end{aligned} \quad (2.21)$$

Once again, I use the subscript ‘ $(X)$ ’ to denote that a parameter or residual is specific to  $X$ . Additionally, I use ‘ $\alpha$ ’ to denote a regression parameter, ‘ $d$ ’ to denote a level-2 residual, and ‘ $r$ ’ to denote a level-1 residual in a covariate model. Similar to  $\theta$ , now  $\phi_X$  includes the multilevel residuals and parameters (i.e.,  $d_{0(X)}$  and  $\sigma_{r(X)}^2$ ). Drawing imputations for  $X$

is a straightforward extension of the previously described Metropolis step (i.e., Equations 2.12, 2.13, and 2.14) using the product of the substantive model’s density and the covariate model’s density; therefore, for the  $i^{th}$  individual within cluster  $j$  the density is as follows.

$$\begin{aligned}
f(x_{ij} \mid y_{ij}, z_j, \theta, \phi_X) &\propto f(y_{ij} \mid x_{ij}, z_j, \theta) f(x_{ij} \mid z_j, \phi_X) \\
&\propto \varphi(y_{ij}; \beta_0 + (\beta_1 + u_{1j})x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0j}, \sigma_e^2) \\
&\quad \times \varphi(x_{ij}; \alpha_{0(X)} + \alpha_{1(X)} z_j + d_{0j(X)}, \sigma_{r(X)}^2)
\end{aligned} \tag{2.22}$$

I have presented Equation 2.22 in a general form with no reference to iteration; however, to clarify, at iteration  $t$  the Metropolis step would evaluate Equation 2.22 at  $\theta = \theta^{(t)}$  and  $\phi_X = \phi_X^{(t)}$  for both the trial imputation and the previous iteration’s imputation.

### Imputing Level-2 Variables

Because  $Z$  is observed at level-2, the covariate model for  $Z$  is still a linear regression model; nevertheless, the algorithm uses the cluster-mean of  $X$  as the predictor. This cluster-mean could be a manifest or latent cluster-mean. The covariate model for  $Z$  is as follows.

$$\begin{aligned}
z_j &= \alpha_{0(Z)} + \alpha_{1(Z)} \bar{x}_j + d_{j(Z)} \\
d_{j(Z)} &\sim \mathcal{N}(0, \sigma_{d(Z)}^2)
\end{aligned} \tag{2.23}$$

For Equation 2.23, I use a subscript ‘ $(Z)$ ’ to denote that a parameter or residual is specific to  $Z$ . Additionally, I use ‘ $\alpha$ ’ to denote regression parameters and ‘ $d$ ’ to denote the residual. Drawing imputations for a level-2 variable requires a modified density. Because  $Y$  is measured at a lower level, the density must account for all observations of  $Y$  that correspond to the level-2 cluster to which the observation of  $Z$  belongs; therefore, the substantive model’s density is proportional to the product of the densities for all  $Y$  observations within the cluster. More concretely, suppose for cluster  $j$  I have 1 to  $n_j$  observations, then the density

for the  $j^{\text{th}}$  observation of  $Z$  is as follows.

$$\begin{aligned}
f(z_j | y_{ij}, x_{ij}, \theta, \phi_Z) &\propto \prod_{i=1}^{n_j} f(y_{ij} | x_{ij}, z_j, \theta) f(z_j | \bar{x}_j, \phi_Z) \\
&\propto f(z_j | \bar{x}_j, \phi_Z) \times \prod_{i=1}^{n_j} f(y_{ij} | x_{ij}, z_j, \theta) \\
&\propto \varphi(z_j; \alpha_{0(Z)} + \alpha_{1(Z)} \bar{x}_j, \sigma_{d(Z)}^2) \\
&\quad \times \prod_{i=1}^{n_j} \varphi(y_{ij}; \beta_0 + (\beta_1 + u_{1_j}) x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0_j}, \sigma_e^2)
\end{aligned} \tag{2.24}$$

Once again, I have presented Equation 2.24 in a general form with no reference to iteration. To clarify, at iteration  $t$  the Metropolis step used to sample from the target density would evaluate Equation 2.24 at  $\theta = \theta^{(t)}$  and  $\phi_Z = \phi_Z^{(t)}$  for both the trial imputation and the previous iteration's imputation. With all variables now imputed, the algorithm then proceeds to the next iteration and starts the process over again.

## Imputation of Binary Level-2 Variables

Within the model-based imputation framework, Enders et al. (in press) provided a brief account of imputing binary and ordinal level-1 variables using a probit regression (Agresti, 2012; Albert & Chib, 1993; Johnson & Albert, 2006) imputation model. The probit regression model imagines discrete responses as an underlying normally distributed latent variable with thresholds dividing the latent variable into the manifest observations. This section serves to extend Enders et al. (in press) by describing imputation for binary level-2 variables in model-based imputation. Returning to analysis model in Equation 2.19, I will now treat  $Z$  as binary. Because of the flexibility of FCS algorithms, I need only adjust the imputation model for  $Z$ . Model-based imputation uses the following covariate model for a binary  $Z$

$$\begin{aligned}
z_j^* &= \alpha_{0(Z)} + \alpha_{1(Z)} \bar{x}_j + d_{j(Z)} \\
d_{j(Z)} &\sim \mathcal{N}(0, \sigma_{d(Z)}^2 = 1)
\end{aligned} \tag{2.25}$$

where  $Z^*$  is an underlying normally distributed latent variable and the residual variance is fixed to 1 for identification. In addition, the model uses a threshold parameter,  $\kappa$ , that

divides the latent distribution into two categories. In general, the model uses the number of categories – 1 threshold parameters in total. Finally, I fix the first threshold parameter to 0 for identification (i.e., the threshold is redundant with the intercept in Equation 2.25).

Each iteration proceeds by first drawing the  $Z^*$  for the complete cases from a truncated normal distribution; therefore, when the observed  $Z$  value is equal to zero, the algorithm samples below the threshold. Conversely, when the observed value is equal to one, the algorithm samples above the threshold. Next, the algorithm then applies the Bayesian sampling steps using  $Z^*$  instead of the observed  $Z$ ; however, because the residual variance is fixed to unity, the residual variance is known and does not need to be sampled. Once all unknown parameters are drawn, trial imputations are sampled on the  $Z^*$  metric and then categorized. The categorized trial imputations are obtained by checking if the  $Z^*$  is above or below the threshold value (i.e., 0 in the binary case). In order to draw imputations, a Metropolis step is employed using the following density to calculate the importance ratio.

$$\begin{aligned}
f(z_j | y_{ij}, x_{ij}, \theta, \phi_Z) &\propto \prod_{i=1}^{n_j} f(y_{ij} | x_{ij}, z_j, \theta) f(z_j^* | \bar{x}_j, \phi_Z) \\
&\propto f(z_j^* | \bar{x}_j, \phi_Z) \times \prod_{i=1}^{n_j} f(y_{ij} | x_{ij}, z_j, \theta) \\
&\propto \varphi(z_j^*; \alpha_{0(Z)} + \alpha_{1(Z)} \bar{x}_j, 1) \\
&\quad \times \prod_{i=1}^{n_j} \varphi(y_{ij}; \beta_0 + (\beta_1 + u_{1j})x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0j}, \sigma_e^2)
\end{aligned} \tag{2.26}$$

Equation 2.26 makes a subtle change to Equation 2.24. I now evaluate the covariate model’s density<sup>2</sup> in terms of the latent  $Z^*$  instead of the categorized  $Z$ . Note the substantive model’s density still evaluates the categorized  $Z$  because the scaling aligns with the analysis model. The Metropolis step would evaluate density in Equation 2.22 at  $\theta = \theta^{(t)}$ ,  $\phi_Z = \phi_Z^{(t)}$  for both the trial imputation and the trial and the previous iteration’s imputation (the “imputation” includes both the latent form and the categorized form).

---

<sup>2</sup>Note that Johnson and Albert (2006) offers an alternative specification of this density where the latent variable is integrated out.

## CHAPTER 3

### Methods

This section outlines five simulations I used to investigate model-based imputation's ability to accommodate cross-level interactions. My simulations greatly expand the work done by Enders et al. (in press), which primarily focused on random slope models. In addition to evaluating model-based imputation, I also included the multilevel extension of transform-then-impute (i.e., the just another variable, or JAV approach). Simulation 1 and Simulation 2 investigated models where model-based imputation correctly specifies the covariate models. More specifically, Simulation 1 investigated a continuous level-2 covariate, and Simulation 2 investigated a binary level-2 covariate. The other three simulations were intended to investigate how robust model-based imputation is to violations in the covariate models; thus, Simulation 3 investigated non-normality in a level-2 covariate, Simulation 4 investigated a continuous level-1 covariate with a non-normally distributed level-2 residual, and Simulation 5 investigated a continuous level-1 covariate with a non-normally distributed level-1 residual. The remainder of this chapter discusses these five simulation studies in greater detail. I first describe the analysis model used for all simulations. Next, I give an overview of the simulation conditions. After that, I describe the data generating procedure to generate the data for Simulation 1. Then, I describe the modifications required to generate the covariates in the remaining four simulations. Finally, I discuss the implementation and the outcome measures for I use for the simulations

### 3.1 Analysis Model

All five simulations were comprised of a level-1 dependent variable,  $Y$ , a level-1 predictor,  $X$ , and a level-2 predictor,  $Z$ . The rationale for using only three variables is to relate my simulation studies to the work already done on single-level interaction effects. Both predictors were incomplete following an MAR mechanism with  $Y$  as the cause of missingness. That is, observations with higher values of  $Y$  are associated with higher a higher likelihood of being unobserved for  $X$  and  $Z$ . The analysis model used for the  $i^{th}$  individual within cluster  $j$  the model is as follows

$$\begin{aligned}
 y_{ij} &= \beta_0 + \beta_1 x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0j} + u_{1j} x_{ij} + e_{ij} \\
 \begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}_u = \begin{bmatrix} \sigma_{u_0}^2 & \sigma_{u_0 u_1} \\ \sigma_{u_0 u_1} & \sigma_{u_1}^2 \end{bmatrix} \right) \\
 e_{ij} &\sim \mathcal{N}(0, \sigma_e^2)
 \end{aligned} \tag{3.1}$$

where  $y_{ij}$  is the level-1 incomplete dependent variable,  $x_{ij}$  is the level-1 incomplete predictor,  $z_j$  is the level-2 incomplete predictor, and  $x_{ij} z_j$  is the cross-level product term. The intercept is given as  $\beta_0$  and each predictor has a corresponding regression coefficient (denoted with a  $\beta$  and a subscript). Finally, the level-2 residuals (i.e.,  $u_0$  and  $u_1$ ) were distributed multivariate normal with means of 0 and an unstructured covariance matrix.

### 3.2 Overview of Simulation Conditions

For all simulations, I arbitrarily set the population value of  $\beta_0$  to equal 10 and the population means of  $X$  and  $Z$  to equal 0. In addition, I fixed the population total variance of  $X$  and  $Z$  to be 1 (when  $Z$  is continuous) and the population variance of  $Y$  to be 100. For the analysis model random effects, I set the covariance between the random effects (i.e.,  $\sigma_{u_0 u_1}$ ) to be specified such that the correlation is equal to 0.30. I specified the variance of the random slope (i.e.,  $\sigma_{u_1}^2$ ) to account for 0.05 of the proportion of within-cluster outcome variance (Rights & Sterba, 2018). For the correlation structure, I allow the variables to all have a

0.4 correlation with each other, and the product of  $X$  and  $Z$  has a 0.15 correlation with  $Y$ . Each simulation had four between-subjects factors: within-cluster sample size ( $n_j = 5, 25, 50, \text{ and } 100$ ), number of clusters ( $J = 25, 50, \text{ and } 200$ ), intraclass correlation ( $\text{ICC} = 0.10 \text{ and } 0.50$ ), and missing data rate (15%, and 30%). Each study used 2,000 replications for each of the 48 conditions. The conditions were chosen based on a review of guidelines from the literature, published Monte Carlo studies, generalizability to typical social science data, and considerations to methodological interests (e.g., evaluating behavior with small sample size and large sample sizes). For example, Maas and Hox (2005) suggested that within cluster sample of  $n_j = 30$  is the norm for level-1 in educational research; I used  $n_j = 5$  to evaluate how the method performs with small level-1 sample size. Similarly, the  $n_j = 100$  condition served to evaluate the large sample size properties of model-based imputation. For number of level-2 clusters, Maas and Hox (2005) suggested that 50 clusters are common in educational research, and Kreft and de Leeuw (1998) recommend a minimum of 30 clusters. Similar to within-cluster sample size, the 200 clusters condition allowed me to investigate the large sample properties of model-based imputation. Turning to the ICC condition, I chose ICC's that are representative of published research (Gulliford, Ukoumunne, & Chinn, 1999; Hedges & Hedberg, 2007; Murray & Blitstein, 2003); thus, I chose an ICC of 0.10 to constitute a value for cross-sectional data, and an ICC of 0.50 to constitute a value for repeated measures data. Finally, the missing data rates I used serve to represent a moderate amount of missing data (i.e., 15% on both predictors) and a large amount of missing data (i.e., 30% on both predictors). The 15% condition is a threshold when traditional FCS begins to show biases (Enders et al., 2018; Enders, Keller, & Levy, 2017; Grund, Lüdtke, & Robitzsch, 2018) and the 30% condition ought to reveal any biases that may occur. Additionally, I used the complete data estimates in all conditions as a baseline. This is helpful in comparing how the complete-data estimation procedure itself behaves in each condition. Finally, Simulation 2 included a separate condition where I investigated a 20%/80% split and a 50%/50% split of the binary level-2 covariate.

### 3.3 Data Generation

Due to the complex nature of multilevel models and the addition of both a random slope and an interaction, this section discusses how I generated the data for Simulation 1. The same basic process was used for Simulations 2 to 5. I subsequently describe the modifications required in Simulations 2 to 5.

Let the matrix  $\mathbf{X}$  refer to the predictors matrix (i.e., column 1 contains the observations of  $X$ , column 2 contains the observations of  $Z$ , and column 3 contains the product of  $X$  and  $Z$ ),  $R$  be the population correlation matrix, and  $\Sigma$  be the population covariance matrix. To obtain  $\Sigma$ , I assume that the population matrix is made up of two orthogonal parts: a level-1 covariance matrix,  $\Sigma_{L1}$ , and a level-2 covariance matrix,  $\Sigma_{L2}$ . Because they are orthogonal, by definition the following is true.

$$\Sigma = \Sigma_{L1} + \Sigma_{L2} \quad (3.2)$$

To obtain  $\Sigma_{L1}$  and  $\Sigma_{L2}$ , I pre- and post-multiply the population correlation matrix with a diagonal matrix containing the square root of the population variance at the respective level. For  $X$  and  $Y$ , the population variance at any given level is defined by the total variance multiplied by the variance partition for the desired ICC condition (e.g., ICC = 0.1 condition constitutes a partition of 0.9 for level-1 and 0.1 for level-2). For the product term, its variance and covariances were empirically estimated using one-hundred million cases.

To obtain solutions for the population regression coefficients for the predictors (denoted by the vector  $\beta'_X = \begin{bmatrix} \beta_1 & \beta_2 \end{bmatrix}$ ), I use the standard linear regression formula as follows

$$\beta_{\mathbf{X}} = \Sigma_{\mathbf{X}}^{-1} \Sigma_{\mathbf{X}\mathbf{Y}} \quad (3.3)$$

where  $\Sigma_{\mathbf{X}}$  is the covariance matrix of the predictors (i.e.,  $X$ ,  $Z$ , and the  $XZ$  product) and  $\Sigma_{\mathbf{X}\mathbf{Y}}$  is a vector of the covariances of the predictors and  $Y$ . Using the population regression coefficients, I solved for the population residual variance of  $Y$  for level-1

$$\sigma_e^2 = \sigma_{L1(Y)}^2 - \beta'_{\mathbf{X}} \Sigma_{L1(\mathbf{X}Y)} - \sigma_{u_1}^2 \sigma_{L1(X)}^2 \quad (3.4)$$

and the population random intercept variance at level-2

$$\sigma_{u_0}^2 = \sigma_{L2(Y)}^2 - \boldsymbol{\beta}'_{\mathbf{X}} \boldsymbol{\Sigma}_{L12\mathbf{X}Y} - \sigma_{u_1}^2 \sigma_{L2(X)}^2 \quad (3.5)$$

where  $\sigma_{u_1}^2$  is the population random slope variance.

Finally, with all parameters determined or otherwise specified (i.e.,  $\sigma_{(u_1)}^2$  is specified to account for the proportion of within-cluster outcome variance explained by level-1 predictors via random slope variation/covariation to be equal to 0.05), the data can be generated. For the level-1 covariate, I drew the level-1 residual for  $i^{th}$  observation within cluster  $j$  (denoted as  $r_{ij(X)}$ ), and for both predictors I drew the level-2 residuals for the  $j^{th}$  cluster (denoted as the row vector  $\mathbf{d}_j$ ).

$$r_{ij(X)} \sim \mathcal{N}(0, \sigma_{L1(X)}^2) \quad (3.6)$$

$$\mathbf{d}_j = \begin{bmatrix} d_{j(X)} \\ d_{j(Z)} \end{bmatrix} \sim \mathcal{N}_2 \left( 0, \boldsymbol{\Sigma}_{L2(\mathbf{X})} = \begin{bmatrix} \sigma_{L2(X)}^2 & \sigma_{L2(X,Z)} \\ \sigma_{L2(X,Z)} & \sigma_{L2(Z)}^2 \end{bmatrix} \right) \quad (3.7)$$

With the residuals drawn, I then used the following formulas to create X and Z.

$$\begin{aligned} x_{ij} &= d_{j(X)} + r_{ij(X)} \\ z_j &= d_{j(Z)} \end{aligned} \quad (3.8)$$

Turning to the dependent variable, I drew the level-1 residual (denoted as  $e$ ), the level-2 residual (denoted as  $u_0$ ), and the random slope residual (denoted as  $u_1$ ) as follows.

$$\begin{aligned} e_{ij} &\sim \mathcal{N}(0, \sigma_e^2) \\ u_{0_j} &\sim \mathcal{N}(0, \sigma_{u_0}^2) \\ u_{1_j} &\sim \mathcal{N}(0, \sigma_{u_1}^2) \end{aligned} \quad (3.9)$$

With the residuals sampled for the dependent variable, I then created Y using the following regression equation.

$$y_{ij} = \beta_0 + \beta_1 x_{ij} + \beta_2 z_j + \beta_3 x_{ij} z_j + u_{0_j} + u_{1_j} x_{ij} + e_{ij} \quad (3.10)$$

Below I have provided the analytical population parameters for the substantive model for the ICC = 0.10 and ICC = 0.50 conditions.

ICC = 0.10

$$y_{ij} = 10 + 3.47(x_{ij}) + 0.77(z_j) + 1.49(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 6.36 & 1.69 \\ 1.69 & 5.0 \end{bmatrix} \right) \quad (3.11)$$

$$e_{ij} \sim \mathcal{N}(0, 69.80)$$

ICC = 0.50

$$y_{ij} = 10 + 3.48(x_{ij}) + 1.84(z_j) + 1.44(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 33.2 & 3.87 \\ 3.87 & 5.0 \end{bmatrix} \right) \quad (3.12)$$

$$e_{ij} \sim \mathcal{N}(0, 40.41)$$

All values have been rounded to three decimal places. Moving forward, I will talk about modifications to the above data generation process for the remaining four simulations. The vast majority of the modifications are done on Equations 3.6, 3.7, and 3.8; therefore, I will write the simulation's respective version of these equations, even if they remain unchanged.

### 3.3.1 Covariate model for Simulation 2

For Simulation 2, I used a binary level-2 covariate instead of a continuous one; thus, Simulation 2's covariate models are as follows.

$$r_{ij(X)} \sim \mathcal{N}(0, \sigma_{L1(X)}^2) \quad (3.13)$$

$$\begin{bmatrix} d_{j(X)} \\ d_{j(Z)} \end{bmatrix} \sim \mathcal{N}_2 \left( 0, \Sigma_{L2(\mathbf{X})} = \begin{bmatrix} \sigma_{L2(X)}^2 & \sigma_{L2(X,Z)} \\ \sigma_{L2(X,Z)} & 1.0 \end{bmatrix} \right) \quad (3.14)$$

$$\begin{aligned}
x_{ij} &= d_{j(X)} + r_{ij(X)} \\
z_j^* &= d_{j(Z)} \\
z_j &= \begin{cases} 0 & \text{if } z_j^* \leq \tau \\ 1 & \text{if } z_j^* > \tau \end{cases}
\end{aligned} \tag{3.15}$$

For the data generation of Simulation 2, Equation 3.14 and 3.15 replaced Equation 3.7 and 3.8. I fixed the variance of the latent  $Z^*$  to 1. Values of  $\tau$  were adjusted dependent on the 80/20 split condition or the 50/50 split condition. Because  $Z^*$  is normally distributed with a variance of 1, I used an inverse cumulative normal density function to calculate the thresholds as  $\tau = -0.842$  and  $\tau = 0.0$  for the respective splits. Unlike the other simulations, for the imputation models,  $Z$  was treated as a binary variable. The distributional assumptions for the covariates map onto the distributional assumptions that model-based imputation requires; therefore, the model-based imputation procedure's models are properly specified. Finally, because the variance of  $Z$  is now dependent on the proportional split, the population parameter values for Simulation 2 are as follows.

ICC = 0.10; 80/20 split

$$\begin{aligned}
y_{ij} &= 10 + 4.16(x_{ij}) + 1.46(z_j) - 1.05(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij} \\
\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 7.25 & 1.81 \\ 1.81 & 5.0 \end{bmatrix} \right) \\
e_{ij} &\sim \mathcal{N}(0, 71.30)
\end{aligned} \tag{3.16}$$

ICC = 0.50; 80/20 split

$$\begin{aligned}
y_{ij} &= 10 + 4.17(x_{ij}) + 3.69(z_j) - 2.17(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij} \\
\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 36.7 & 4.06 \\ 4.06 & 5.0 \end{bmatrix} \right) \\
e_{ij} &\sim \mathcal{N}(0, 40.16)
\end{aligned} \tag{3.17}$$

ICC = 0.10; 50/50 split

$$\begin{aligned}
y_{ij} &= 10 + 5.84(x_{ij}) + 1.22(z_j) - 3.81(x_{ij}z_j) + u_{0_j} + u_{1_j}x_{ij} + e_{ij} \\
\begin{bmatrix} u_{0_j} \\ u_{1_j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 6.76 & 1.74 \\ 1.74 & 5.0 \end{bmatrix} \right) \\
e_{ij} &\sim \mathcal{N}(0, 68.28)
\end{aligned} \tag{3.18}$$

ICC = 0.50; 50/50 split

$$\begin{aligned}
y_{ij} &= 10 + 5.69(x_{ij}) + 2.85(z_j) - 4.02(x_{ij}z_j) + u_{0_j} + u_{1_j}x_{ij} + e_{ij} \\
\begin{bmatrix} u_{0_j} \\ u_{1_j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 34.5 & 3.94 \\ 3.94 & 5.0 \end{bmatrix} \right) \\
e_{ij} &\sim \mathcal{N}(0, 38.72)
\end{aligned} \tag{3.19}$$

### 3.3.2 Covariate model for Simulation 3

For Simulation 3, I generated a non-normal level-2 covariate. Recall, in Equations 2.24 and 2.23 I assume that  $Z$  is normally distributed; therefore, I checked the robustness of this assumption by generating  $Z$  to follow a  $\chi^2(1)$  distribution. This results in the following models for the covariates  $X$  and  $Z$ .

$$r_{ij(X)} \sim \mathcal{N}(0, \sigma_{L1(X)}^2) \tag{3.20}$$

$$\begin{bmatrix} d_{j(X)} \\ d_{j(Z)} \end{bmatrix} = \begin{bmatrix} d_{j(X)}^* & \frac{d_{j(Z)}^* - 1}{\sqrt{2}} \end{bmatrix} \mathbf{L} \tag{3.21}$$

$$d_{j(X)}^* \sim \mathcal{N}(0, 1)$$

$$d_{j(Z)}^* \sim \chi^2(1)$$

$$x_{ij} = d_{j(X)} + r_{ij(X)} \tag{3.22}$$

$$z_j = d_{j(Z)}$$

For data generation in Simulation 3, Equation 3.21 replaced Equation 3.7. In Equation 3.21,  $\mathbf{L}$  is the Cholesky decomposition of  $\Sigma_{L2(\mathbf{X})} = \begin{bmatrix} \sigma_{L2(X)}^2 & \sigma_{L2(X,Z)} \\ \sigma_{L2(X,Z)} & \sigma_{L2(Z)}^2 \end{bmatrix}$ . To generate the data I first drew  $d_{j(Z)}^*$  from a chi-square distribution with one degree of freedom and then scaled it to have a mean of 0 and a variance of 1. Next, I took the Cholesky decomposition of the  $\Sigma_{L2(\mathbf{X})}$  matrix and post-multiply it to correlate the residuals and scale the variances. Finally, the population parameters for the two ICC conditions are as follows.

ICC = 0.10

$$y_{ij} = 10 + 3.82(x_{ij}) + 0.65(z_j) + 0.90(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 5.79 & 1.61 \\ 1.61 & 5.0 \end{bmatrix} \right) \quad (3.23)$$

$$e_{ij} \sim \mathcal{N}(0, 71.67)$$

ICC = 0.50

$$y_{ij} = 10 + 3.63(x_{ij}) + 1.90(z_j) - 0.29(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 34.9 & 3.96 \\ 3.96 & 5.0 \end{bmatrix} \right) \quad (3.24)$$

$$e_{ij} \sim \mathcal{N}(0, 40.87)$$

### 3.3.3 Covariate model for Simulation 4

For Simulation 4, I generated a non-normal level-2 residual for the level-1 covariate. Recall, in Equations 2.21 and 2.22 I assume that the level-2 residual for  $X$  is normally distributed; thus, I checked the robustness of this assumption by generating  $X$ 's level-2 residual to follow a  $\chi^2(1)$  distribution. This results in the following models for the covariates  $X$  and  $Z$ .

$$r_{ij(X)} \sim \mathcal{N}(0, \sigma_{L1(X)}^2) \quad (3.25)$$

$$\begin{bmatrix} d_{j(X)} \\ d_{j(Z)} \end{bmatrix} = \begin{bmatrix} \frac{d_{j(X)}^* - 1}{\sqrt{2}} & d_{j(Z)}^* \end{bmatrix} \mathbf{L} \quad (3.26)$$

$$d_{j(X)}^* \sim \chi^2(1)$$

$$d_{j(Z)}^* \sim \mathcal{N}(0, 1)$$

$$x_{ij} = d_{j(X)} + r_{ij(X)} \quad (3.27)$$

$$z_j = d_{j(Z)}$$

Similar to Simulation 3,  $\mathbf{L}$  is the Cholesky decomposition of  $\Sigma_{L2(\mathbf{x})} = \begin{bmatrix} \sigma_{L2(X)}^2 & \sigma_{L2(X,Z)} \\ \sigma_{L2(X,Z)} & \sigma_{L2(Z)}^2 \end{bmatrix}$ .

Simulation 4 follows the same logic as Simulation 3, except that I am now treating the level-2 residual for  $X$  as chi-square distributed instead of the residual of  $Z$ ; therefore, Equation 3.26 replaces Equation 3.7. Finally, the population parameters for the two ICC conditions are as follows.

ICC = 0.10

$$y_{ij} = 10 + 3.90(x_{ij}) + 0.77(z_j) + 1.49(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 6.36 & 1.69 \\ 1.69 & 5.0 \end{bmatrix} \right) \quad (3.28)$$

$$e_{ij} \sim \mathcal{N}(0, 69.80)$$

ICC = 0.50

$$y_{ij} = 10 + 3.48(x_{ij}) + 1.85(z_j) + 1.44(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij}$$

$$\begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} \sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 33.2 & 3.87 \\ 3.87 & 5.0 \end{bmatrix} \right) \quad (3.29)$$

$$e_{ij} \sim \mathcal{N}(0, 40.41)$$

### 3.3.4 Covariate model for Simulation 5

For Simulation 5, I generated a non-normal level-1 residual for the level-1 covariate. Recall, in Equations 2.21 and 2.22 I assume that the level-1 residual for  $X$  is normally distributed; thus, I checked the robustness of this assumption by generating  $X$ 's level-1 residual to follow a  $\chi^2(1)$  distribution. This results in the following models for the covariates  $X$  and  $Z$ .

$$\begin{aligned} r_{ij(X)}^* &\sim \chi^2(1) \\ r_{ij(X)} &= \sigma_{L1(X)}^2 \left( \frac{r_{ij(X)}^* - 1}{\sqrt{2}} \right) \end{aligned} \quad (3.30)$$

$$\begin{bmatrix} d_{j(X)} \\ d_{j(Z)} \end{bmatrix} \sim \mathcal{N}_2 \left( 0, \Sigma_{L2(\mathbf{X})} = \begin{bmatrix} \sigma_{L2(X)}^2 & \sigma_{L2(X,Z)} \\ \sigma_{L2(X,Z)} & \sigma_{L2(Z)}^2 \end{bmatrix} \right) \quad (3.31)$$

$$\begin{aligned} x_{ij} &= d_{j(X)} + r_{ij(X)} \\ z_j &= d_{j(Z)} \end{aligned} \quad (3.32)$$

Similar to the previous simulations, I transformed the chi-square residual (i.e.,  $r_{ij(X)}^*$ ) such that it has a variance of 1 and mean of 0 and then multiply it by the appropriate variance (i.e.,  $\sigma_{L1(X)}^2$ ) to obtain  $r_{ij(X)}$ ; therefore, Equation 3.30 serves to replace Equation 3.6. Finally, the population parameters for the two ICC conditions are as follows.

ICC = 0.10

$$\begin{aligned} y_{ij} &= 10 + 3.47(x_{ij}) + 0.77(z_j) + 1.49(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij} \\ \begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 6.36 & 1.69 \\ 1.69 & 5.0 \end{bmatrix} \right) \\ e_{ij} &\sim \mathcal{N}(0, 69.80) \end{aligned} \quad (3.33)$$

ICC = 0.50

$$\begin{aligned}
 y_{ij} &= 10 + 3.48(x_{ij}) + 1.84(z_j) + 1.44(x_{ij}z_j) + u_{0j} + u_{1j}x_{ij} + e_{ij} \\
 \begin{bmatrix} u_{0j} \\ u_{1j} \end{bmatrix} &\sim \mathcal{N}_2 \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_u = \begin{bmatrix} 33.2 & 3.87 \\ 3.87 & 5.0 \end{bmatrix} \right) \\
 e_{ij} &\sim \mathcal{N}(0, 40.41)
 \end{aligned} \tag{3.34}$$

### 3.4 Simulation Implementation and Outcome Measures

To implement the simulations, I used several software packages. For data generation, I implemented my described process in R programming language for statistical computing (R Core Team, 2018) and generated 2000 replications per condition. To impute the data for both model-based imputation and transform-then impute (using a traditional FCS procedure), I used the Blimp software package (Keller & Enders, 2019). I generated ten imputations per replication. And used a burn-in interval of 5000 and a thinning interval of 1000; however, I used a sample of replications across conditions to assess if these values need to be changed. To analyze the data, I used *Mplus* statistical software package and then used R code to pool the parameter estimates and standard errors. Finally, using R, I computed the outcome measures described below on these pooled parameter estimates and standard errors.

For all five simulations, I focused on two outcome measurements: percent bias and confidence interval (C. I.) coverage. C. I. coverage was only calculated on the fixed effects because the literature suggests that symmetric normal-theory confidence intervals are inappropriate for variance parameters (Maas & Hox, 2005; Snijders & Bosker, 2012). I define percent bias as

$$\text{percent bias} = \frac{(\text{average estimate}) - (\text{true parameter})}{(\text{true parameter})} \times 100 \tag{3.35}$$

and confidence interval coverage as

$$95\% \text{ C. I. coverage} = \frac{(\text{number of replications with population parameter in C. I.})}{(\text{total replications})} \tag{3.36}$$

These measures were calculated for all simulations and conditions. I used these measure-

ments to compare transform-then-impute, model-based imputation, and the complete-data estimates. As mentioned previously, often the estimation procedure for the analysis model itself contains some bias and the complete-data estimates served as a baseline to compare the other methods. For my investigation, I used some rules of thumb to assess the practical implications of bias. More specifically, for percent bias the literature recommends that the bias does not exceed 10% (Kaplan, 1988), and for C.I. coverage, Bradley (1978) suggested a “liberal” criterion for 95% C. I. coverage, with a lower limit of 0.925 and an upper limit of 0.975.

# CHAPTER 4

## Results

### 4.1 Simulation 1 – Normally Distributed Covariates

Figure 1 and 2 are trellis plots displaying percent bias values (i.e., Equation 3.35) for the  $ICC = 0.1$  and  $ICC = 0.5$  with 30 % missing data rate. I have focused on 30% missing data condition because it amplifies the observed biases across the board, and I provide the full simulation results in Appendix B. Additionally, I provide the listwise deletion results in Appendix G. The absolute values of the listwise deletion biases were consistently above 20%, which suggests that the MAR mechanism induces parameter bias if not properly handled. As a rule of thumb, previously published simulations often use a  $\pm 10\%$  in bias as acceptable (Kaplan, 1988); thus, the figures display these heuristics as dashed lines. Considering Simulation 1's results as a whole, the model-based imputation estimates tracked closely with the complete data estimates in the higher sample size conditions (e.g.,  $L2 = 200$  and  $L1 = 100$ ) and slowly started to deviate as sample size was reduced. The parameter estimates generally stayed acceptable except in the two smallest sample size conditions. In contrast, the JAV was consistently worse than model-based imputation with biases exceeding 20% to 30% in the best conditions. The JAV approach had consistently biased estimates for the random slope variance ( $\sigma_{u_1}^2$ ), the random effects covariance ( $\sigma_{u_0, u_1}$ ), and the regression coefficient for the cross-level interaction ( $\beta_3$ ).

Focusing on model-based imputation, the most problematic conditions were all due to small sample sizes,  $N = 125$  ( $L1 = 5$ ,  $L2 = 25$ ) and  $N = 250$  ( $L1 = 5$  and  $L2 = 50$ ). In these two small sample size conditions, there were observed biases in both the fixed effect parameters and variance parameters (6 of 8 parameters in  $N = 125$  and 4 of 8 parameters in

$N = 250$ ). These biases were slightly reduced with a decrease in the missing data rate and seemed to be mostly unaffected by ICC. The regression coefficient for the level-2 predictor was mainly affected by the level-2 sample size, such that in the  $L2 = 200$  conditions the bias was effectively eliminated. Turning to the regression coefficient for the cross-level interaction, increases in the level-1 and level-2 sample sizes both caused reduction in the observed bias, and this parameter had around 10% bias in both the  $N = 250$  ( $L1 = 5, L2 = 50$ ) and  $N = 625$  conditions ( $L1 = 25, L2 = 25$ ). Looking at the two

$$N =$$

1,250 conditions (i.e.,  $L1 = 25, L2 = 50$  and  $L1 = 50, L2 = 25$ ) illustrates that increasing the level-2 units reduced the biases in both the regression coefficients for the cross-level interaction and level-2 predictor. For example, the regression coefficient for the level-2 predictor dropped from roughly 18% bias to roughly 10% bias with that increase. Moreover, comparing these two conditions between Figure 1 and 2 (i.e., the ICC conditions) we can see that the higher ICC condition also reduced the observed biases in the two regression coefficients by approximately 5%. The intercept variance and level-1 residual variance largely mapped onto the complete data estimates in all conditions. In contrast, the random slope variance and the covariance between the random slope and random intercept both were biased in the smaller sample size conditions; however, these biases quickly approached zero as sample size increased (i.e., the random slope variance is close to zero in the  $N = 625; L1 = 25, L2 = 25$  condition).

Figures 3 and 4 give the confidence interval coverage of the fixed effects for the two ICC conditions (30% missing data rate). The figures include a dashed line at 0.925 and 0.975 coverage to represent acceptable values of coverage (Bradley, 1978). JAV consistently had poor coverage for both the level-1 predictor and the cross-level interaction. The cross-level interaction performed especially poorly. For example, in the  $L1 = 100, L2 = 200, ICC = 0.5$  condition JAV the 95% confidence intervals contained the true parameter only 46% of the time. The poor coverage for this parameter is most likely due to the observed bias in this parameter. In comparison, model-based imputation's coverage was 94% for this same

parameter in the same condition. Overall, model-based imputation’s parameter coverage was within these bounds in all but three of the conditions. The regression coefficient for the level-2 predictor had coverage that fell below 0.925 in three of the small level-2 sampling unit conditions (i.e., L1 = 50, L2 = 25, ICC = 0.5; L1 = 100, L2 = 25, ICC = 0.5; L1 = 100 L2 = 25, ICC = 0.1). However, model-based imputation still outperformed both the complete data and JAV in these conditions. Looking at the large sample size conditions, the coverage for the model-based imputation and the complete data became nearly identical.

## 4.2 Simulation 2 – Level-2 Binary Covariate

Turning to Simulation 2, recall that the level-2 binary variable had two different distributions (50/50 and 80/20); thus, figures 5 through 8 are trellis plots displaying percent bias values for the 30% missing data rate. Similar to Simulation 1, I have focused on 30% missing data condition because it amplifies the observed parameter biases, and I provide the 15% missing data rate results in Appendix C. The absolute values of the listwise deletion biases were consistently above 20%, which suggests that the MAR mechanism was strong enough to induce parameter bias if not appropriately handled (see Appendix H for plots). Looking at the JAV parameter bias, the approach performed worse than model-based imputation. For example,  $\beta_3$  consistently exceeded 40% bias in the large sample size conditions. Moreover, JAV also had biased estimates for the random slope variance ( $\sigma_{u_1}^2$ ), random effects covariance ( $\sigma_{u_0, u_1}$ ), and the regression coefficient for the cross-level interaction. These biases in JAV are present across all sample sizes and only amplified by the smaller sample sizes.

Consistent with Simulation 1, the most problematic conditions for model-based imputation were all due to small sample sizes,  $N = 125$  (L1 = 5, L2 = 25) and  $N = 250$  (L1 = 5 and L2 = 50). In these two small sample size conditions, there were observed biases in both the fixed effect parameters and variance parameters (5 to 6 parameters in  $N = 125$  and 4 of 8 parameters in  $N = 250$ ). These biases were slightly reduced with a decrease in missing data rate and were mostly unaffected by ICC. The regression coefficient for the level-2 binary predictor was mainly affected by the level-2 sample size and the percentage of observations

in each category. For example, in the  $L2 = 200$  condition, the parameter bias was effectively eliminated in the 50/50 distribution but was just below 10% in the 80/20 condition. Overall, there was often about a five to ten percent drop in bias for the level-2 regression coefficient when moving to a 50/50 split. This suggests that larger sample sizes are needed to obtain unbiased estimates of the probit regression's threshold parameters. For example, in the  $L2 = 25$  and  $50$  conditions we might expect there to be few observed cases in the 20% category.<sup>1</sup> Turning to the regression coefficient for the cross-level interaction, increases in the level-1 and level-2 sample sizes independently reduced the observed parameter bias. The distribution condition did not play much of a role in this bias, and the coefficient for the cross-level interaction quickly dropped under 10% bias as sample size increased (e.g., by  $N = 625$ ;  $L1 = 25$ ,  $L2 = 25$ ). The intercept variance and level-1 residual variance largely mapped onto the complete data estimates in all conditions. In contrast, the random slope variance was biased in the two smallest size conditions; however, this bias quickly approached zero as sample size increased (i.e., the random slope variance is close to zero in the  $N = 625$ ;  $L1 = 25$ ,  $L2 = 25$  condition). Looking at the covariance of the random effects, this parameter remained biased across conditions until the level-2 sampling units were in the highest condition.

Figures 9 through 12 give the confidence interval coverage of the fixed effects for the four ICC and percentage split combinations (30% missing data rate). Overall, the confidence interval coverage for model-based imputation almost always stayed between the bounds defined by Bradley (1978). The main exceptions were seen in the 80/20 split condition for both ICCs. The regression coefficients for the level-2 binary variable and the cross-level interaction had poor coverage in two low level-2 sample size conditions (i.e.,  $L1 = 50$ ,  $L2 = 25$ ;  $L1 = 100$ ,  $L2 = 25$  in Figures 10 and 12). However, the coverage in these conditions matched closely to the corresponding complete data coverage. JAV consistently had poor coverage for both the level-1 predictor and the cross-level interaction. The cross-level interaction especially performed poorly. Across all four figures, the coverage for JAV of both these coefficients consistently did worse as sample size increased.

---

<sup>1</sup>This is also dependent on the selection mechanism. For example, the missing observations were generated in such a way that the 20% category was had higher missing data rate than the 80% category.

### 4.3 Simulation 3 – Non-normal Level-2 Covariate

For Simulation 3, recall that the residual of the level-2 covariate was distributed based on a chi-square with one degree of freedom distribution. Figure 13 and 14 are trellis plots displaying percent bias values for the 30% missing data rate. Similar to the previous simulations, I have focused on 30% missing data condition because it amplifies the observed parameter biases, and I provide the full simulation results in Appendix D. The absolute values of the listwise deletion biases were consistently above 20%, which suggests that the MAR mechanism was strong enough to induce bias if not appropriately handled (see Appendix I for plots). Similar to the previous simulations, the JAV approach produced consistently biased estimates of the random slope variance ( $\sigma_{u_1}^2$ ), random effects covariance ( $\sigma_{u_0, u_1}$ ), and the regression coefficient for the cross-level interaction ( $\beta_3$ ). For example, the  $\beta_3$  regression coefficient consistently ranged from 20% to 40% bias across all sample sizes with 30% missing data rate.

Considering Simulation 3's results as a whole, the model-based imputation estimates tracked closely with the complete data estimates in the highest sample size conditions (i.e.,  $L2 = 200$  and  $L1 = 100$ ) with biases between  $\pm 5\%$  for all parameters. These biases started to deviate as both the number of level-1 sampling units and level-2 sampling units were reduced. The most problematic conditions were all due to small sample sizes,  $N = 125$  ( $L1 = 5$ ,  $L2 = 25$ ) and  $N = 250$  ( $L1 = 5$  and  $L2 = 50$ ). In these two small sample size conditions, there were observed biases in both the fixed effect parameters and variance parameters (6 of 8 parameters in  $N = 125$  and 4 of 8 parameters in  $N = 250$ ). As with previous simulations, JAV performed just as poorly in these conditions.

For both methods, the intercept variance and level-1 residual variance largely mapped onto the complete data estimates in all conditions. Focusing on the random slope variance ( $\sigma_{u_1}^2$ ), model-based imputation's estimate was biased in the two small sample size conditions, but this bias quickly approached zero by the  $N = 625$  ( $L1 = 25$ ,  $L2 = 25$ ) condition. In contrast, JAV had consistent bias in this parameter (e.g., the parameter was underestimated by roughly 20% in the largest sample size condition with  $ICC = 0.1$ ). Excluding the two smallest

sample sizes in the ICC = 0.1 condition, the model-based imputation estimate the covariance between the random effects ( $\sigma_{u_0, u_1}$ ) closely mapped onto the complete data estimate. Looking at JAV, this parameter consistently performed poorly (i.e., biases consistently worse than  $-20\%$ ). Increasing the ICC to 0.5 also increased the bias in  $\sigma_{u_0, u_1}$  for both methods; however, model-based imputation parameter estimate eventually reached approximately 0% bias as the number of level-2 units increased to 200. In contrast, JAV remained consistently underestimated.

Turning to the regression coefficients, the two regression coefficients associated with the misspecification (i.e.,  $\beta_2$  and  $\beta_3$ ) performed worse than Simulation 1. The bias persisted across both ICC conditions and was amplified by increasing the missing data rate. For the level-2 predictor's regression coefficient ( $\beta_2$ ) the bias was attenuated as the number of level-2 sample units increased for both methods. However, model-based imputation consistently performed better than JAV. For example, in the L1 = 25, L2 = 25, ICC = 0.5 condition (Figure 15) model-based imputation was approximately  $-10\%$  and JAV was approximately  $-20\%$  bias. For the cross-level interaction regression coefficient ( $\beta_3$ ), the sample size conditions was the main factor in reducing the bias. Lowering the ICC also helped this parameter for both methods. When comparing JAV and model-based imputation, model-based imputation consistently outperformed JAV for  $\beta_3$ . JAV consistently overestimated this parameter with biases ranging from 20% to greater than 40%. In contrast, model-based imputation's bias was reduced as sample size increased. In the largest sample size condition, the bias was approximately 3% (ICC = 0.1) to 5% (ICC = 0.5).

Figures 15 and 16 give the confidence interval coverage of the fixed effects for the two ICC conditions (30% missing data rate). The figures include a dashed line at 0.925 and 0.975 coverage. Overall, model-based imputation's parameter coverage was within these bounds in all but three of the conditions. The regression coefficient for the level-2 predictor had coverage that fell below 0.925 in the three of the small level-2 sample size conditions (i.e., L1 = 25, L2 = 25, ICC = 0.5; L1 = 100, L2 = 25, ICC = 0.5; L1 = 100 L2 = 25, ICC = 0.1) and one medium level-2 sampling unit condition (i.e., L1 = 5, L2 = 50, ICC = 0.5). However, model-

based imputation still outperformed both the complete data and JAV in these conditions. Looking at the large sample size conditions, model-based imputation and the complete data became nearly identical. In contrast, JAV consistently had poor coverage for both the level-1 predictor and the cross-level interaction. The cross-level interaction performed especially poorly. For example, in the  $L1 = 100$ ,  $L2 = 200$ ,  $ICC = 0.5$  condition the 95% confidence intervals contained the true parameter only 16% of the time for JAV. In comparison, model-based imputation's coverage was 94.5% for this parameter in the same condition.

#### 4.4 Simulation 4 – Level-1 Covariate with Non-normal Level-2 Residual

For Simulation 4, recall that the level-2 residual for  $X$  was distributed as a  $\chi^2(1)$  distribution. Figure 17 and 18 are trellis plots displaying percent bias values for the 30% missing data rate. Similar to the previous simulations, I have focused on 30% missing data condition because it amplifies the observed parameter biases, and I provide the full simulation results in Appendix E. The absolute values of the listwise deletion biases were consistently above 20%, which suggests that the MAR mechanism was strong enough to induce bias if not appropriately handled (see Appendix J for plots). Considering Simulation 4's results as a whole, the model-based imputation estimates tracked closely with the complete data estimates in the higher sample size conditions (e.g.,  $L2 = 200$  and  $L1 = 100$ ) and slowly started to deviate as sample size was reduced. The parameter estimates were generally acceptable except in the two smallest sample size conditions. The regression coefficient for the level-1 predictor ( $\beta_1$ ) did not appear to be affected by the misspecification in Simulation 4. In contrast, JAV remained consistently biased in its estimates of the random slope variance ( $\sigma_{u_1}^2$ ), random effects covariance ( $\sigma_{u_0, u_1}$ ), and the regression coefficient for the cross-level interaction ( $\beta_3$ ). Similar to previous simulations, these biases in JAV were present across all sample sizes and only amplified by the smaller sample size conditions.

Focusing on model-based imputation, the most problematic conditions were all due to small sample sizes,  $N = 125$  ( $L1 = 5$ ,  $L2 = 25$ ) and  $N = 250$  ( $L1 = 5$  and  $L2 = 50$ ). In these two small sample size conditions, there were observed biases in both the fixed effect parameters and variance parameters (6 of 8 parameters in  $N = 125$  and 4 of 8 parameters in  $N = 250$ ). These biases were slightly reduced with a decrease in missing data rate. The regression coefficient for the level-2 predictor was mainly affected by the level-2 sample size, such that in the  $L2 = 200$  condition the bias is effectively eliminated. Turning to the regression coefficient for the cross-level interaction, increases in the level-1 and level-2 sample sizes reduced the observed bias, and this parameter performed adequately in both the  $N = 250$  ( $L1 = 5$ ,  $L2 = 50$ ) and  $N = 625$  conditions ( $L1 = 25$ ,  $L2 = 25$ ). Looking at both  $N = 1,250$  conditions (i.e.,  $L1 = 25$ ,  $L2 = 50$  and  $L1 = 50$ ,  $L2 = 25$ ) illustrates the effects of having an increase in level-2 units reduces the biases in the regression coefficients for the cross-level interaction and level-2 predictor. Moreover, comparing these two conditions between Figure 17 and 18 (i.e., the ICC conditions) we can see that the higher ICC condition also reduced the observed biases in the regression coefficients for both cross-level interaction and level-2 predictor. The intercept variance and level-1 residual variance largely mapped onto the complete data estimates in all conditions. In contrast, the random slope variance and the covariance between the random slope and random intercept were biased in the smaller sample size conditions. The bias in the covariance quickly approach zero as sample size increased (i.e., the covariance is close to zero in the  $N = 625$ ;  $L1 = 25$ ,  $L2 = 25$  condition). The slope variance also fell under 10% bias when sample size increased (e.g.,  $N = 625$ ;  $L1 = 25$ ,  $L2 = 25$ ) and was more affected by increasing the number of within cluster sampling units than between cluster sampling units. The higher ICC condition led to an 8% bias in the slope variance that persisted even in the highest sampling size condition (i.e.,  $N = 20,000$ ;  $L1 = 100$ ,  $L2 = 200$ ). Decreasing the ICC to 0.1 had a small reduction of this bias (approximately 3%).

Figures 19 and 20 give the confidence interval coverage of the fixed effects for the two ICC conditions (30% missing data rate). The figures include a dashed line at 0.925 and

0.975 coverage. Overall, model-based imputation’s coverage was within these bounds in all but three of the conditions. Compared to previous simulations, the misspecification in the level-2 residual of  $X$  had slightly worse coverage of the cross-level interaction. Specifically, for model-based imputation, the regression coefficient for the cross-level interaction’s coverage fell below 0.925 in the three of the small level-2 sampling unit conditions (i.e.,  $L1 = 50, L2 = 25, ICC = 0.5$ ;  $L1 = 50, L2 = 25, ICC = 0.1$ ;  $L1 = 100, L2 = 25, ICC = 0.1$ ) and one medium level-2 sampling unit condition (i.e.,  $L1 = 50, L2 = 50, ICC = 0.5$ ). However, model-based imputation still outperformed JAV. Looking at the large N conditions, model-based imputation and the complete data were nearly identical. In contrast, JAV consistently had poor coverage for both the level-1 predictor and the cross-level interaction. The cross-level interaction performed especially poorly. For example, in the  $L1 = 100, L2 = 200, ICC = 0.5$  condition the 95% confidence intervals contained the true parameter only 38% of the time for JAV. In comparison, model-based imputation’s coverage was 94% for this parameter in the same condition.

#### **4.5 Simulation 5 – Level-1 Covariate with Non-normal Level-1 Residual**

Figure 21 and 22 are trellis plots displaying percent bias values for the 30% missing data rate. Similar to the previous simulations, I have focused on 30% missing data condition because it amplifies the observed parameter biases, and I provide the full simulation results in Appendix F. The absolute values of the listwise deletion biases were consistently above 40% in the 30% missing data rate condition, which suggests that the MAR mechanism was strong enough to induce bias if not appropriately handled (see Appendix F for plots). Considering Simulation 5’s results as a whole, both methods had substantial bias in several parameters across all conditions. The JAV approach produced consistently biased estimates of the random slope variance ( $\sigma_{u_1}^2$ , ranging from 50% to 200%), random effects covariance ( $\sigma_{u_0, u_1}$ , ranging  $-50\%$  to  $-100\%$ ) and the regression coefficient for the cross-level interaction ( $\beta_3$ , ranging 50%

to 100%). Model-based imputation estimates also had substantial bias in the random slope variance (ranging from approximately 80% to 200% depending on the condition). In addition, for model-based imputation there was observed bias in the cross-level interaction regression coefficient (ranging approximately between 3% to 30%), but this was substantially less than JAV (ranging approximately between 50% to 100%).

Because of the large bias observed in Simulation 5, I further investigated the simulation. To begin, I looked at scatterplots of  $Y$  regressed on  $X$  by cluster. Figure 23 is a sample of twelve clusters ( $ICC = 0.5$ , 30% missing data rate,  $L1 = 250$ ,  $L2 = 50$ ). In each scatterplot, I plotted the complete data (represented by a circle) and one imputed data set (represented by a plus), with regression lines for each within cluster regression (solid for complete, dashed for imputations). Note that if the circle and plus overlap then the observation would have been observed in the imputed data set, and if the plus is offset horizontally, then this is an imputation ( $Y$  is always complete so there will be no vertical offset). Figure 23 illustrates the extreme floor effect caused by the high skew of the  $\chi^2(1)$  distribution. Moreover, the figure also illustrates that the missing data selection mechanism was defined such that higher values of  $Y$  were more likely to be missing observations in  $X$ . For example, looking at cluster 37 in Figure 23 we see that the bulk of missing values (i.e., when the circle and plus points do not overlap) are on the upper tail of  $X$ . These observations exert a high influence on the cluster-specific linear regression, and the assumption of a normally distributed residual does not adequately produce imputations for these observations with high influence. Thus, while the average/fixed regression slope may not be as affected, the variance of these cluster-specific slopes is overestimated because high-leverage data points are getting deleted, and imputation is replacing these scores with low-leverages points in a different part of the distribution.

To investigate the selection mechanism further, I ran a follow up simulation ( $ICC = 0.5$ , missing data rate = 30%,  $L1 = [25, 50, 100]$ ,  $L2 = [50, 100]$ ), reversing the relationship between  $Y$  and  $X$ . By making  $Y$  and  $X$  negatively correlated, the MAR mechanism caused values on the tail of  $X$  to be more likely to be observed. Figure 24 is the percent bias values for the follow up simulation. Comparing Figure 24 to Figure 22, one can see the magnitude of the

percent bias in the cross-level interaction and random slope variance decreased drastically. In addition, the direction of the bias also reversed. This highlights that it is important to also consider functional relationship of the MAR mechanism when looking at highly skewed data.

Table 4.1: Percent bias of posterior means for varying degrees of freedom.

$\chi^2$ $df$	1	5	10	15	20	25	50
Skewness	2.83	1.26	0.89	0.73	0.63	0.57	0.40
Ex. Kurtosis	12.00	2.40	1.20	0.80	0.60	0.48	0.24
$\beta_0$	2.76	1.14	0.64	0.56	-0.65	1.11	-0.43
$\beta_1$	1.57	-1.13	2.98	-0.64	-1.16	2.26	-0.81
$\beta_2$	-17.76	-20.69	-12.50	-9.14	-10.71	-11.58	-8.00
$\beta_3$	25.69	9.82	7.67	10.23	3.86	5.87	2.12
$\sigma_{u_0}^2$	20.55	15.75	13.16	2.43	5.13	4.42	5.56
$\sigma_{u_1}^2$	86.32	30.64	18.32	10.84	11.90	11.66	5.10
$\sigma_{u_0, u_1}$	-86.55	-44.57	-29.49	-27.44	-12.08	-2.17	-9.59
$\sigma_e^2$	2.75	1.60	1.10	0.99	0.88	0.60	0.50

*Note:* Skewness and excess kurtosis are computed for the level-1 residual.

Returning to Simulation 5, because the observed biases were present even in the largest sample size condition, I decided to investigate the asymptotic properties. I generated large sample size data sets ( $N = 1,000,000$ ;  $L1 = 250$ ,  $L2 = 4000$ ) for the  $ICC = 0.5$  and 30% missing data rate. I then varied the degrees of freedom for the chi-square distribution (i.e.,  $df = 1, 5, 10, 15, 20, 25$ , and 50). With these large sample size data sets, I used the model-based imputation procedure to obtain Bayesian posterior means of the parameters. Table 4.1 provides the percent bias of the posterior means for each parameter as the degrees of freedom change. Although this single replication simulation is susceptible to sampling error, there is a clear pattern of decreasing bias as the data become less skewed and kurtotic. For example, the random slope variance's bias quickly drops from 80% to 10% when the chi-square has 15

degrees of freedom (residual's skewness = 0.73; residual's excess kurtosis = 0.8). Similar to the previous scatterplot (i.e., Figure 23), Figure 25 is a scatterplot of  $Y$  regressed on  $X$  for twelve clusters. This figure illustrates that the data do not have such an extreme floor effect but instead starts to approach normality. These findings suggest that, while the extreme and possibly unrealistic cases of a  $\chi^2(1)$  distribution within cluster residual produced large amounts of bias, that bias was reduced as the data became more normal.

## CHAPTER 5

### Discussion

A large body of research supports the use of multiple imputation as a method to handle missing data. When using multiple imputation, it is important that the “filled-in” values are generated from a model that is at least as rich as the researcher’s analysis model. This is especially the case when the analysis models include interactions. Some of the original ad-hoc methods proposed for interactions, such as imputing the product terms, have been shown to produce biased estimates under a MAR assumption (Enders et al., 2014; Seaman et al., 2012). A growing body of recent research has suggested using fully Bayesian multiple imputation methods (Bartlett et al., 2014; Erler, Rizopoulos, Jaddoe, Franco, & Lesaffre, 2019; Erler et al., 2016; Goldstein et al., 2014; Kim, Belin, & Sugar, 2018; Kim, Sugar, & Belin, 2015; Zhang & Wang, 2017). The fully Bayesian multiple imputation methods require direct specification of the analysis model to ensure that the imputations map onto it (i.e., are compatible with the analysis).

Building on the developments of Enders et al. (in press), this dissertation has investigated a model-based imputation framework that should appropriately handle interactions in a multilevel setting. In the model-based framework, imputations for the incomplete covariates are drawn from distributions comprised of two separate models, the analysis model (i.e., the model that a researcher is interested in estimating) and a covariate model. This dissertation investigated the use of this model-based imputation framework in the context of cross-level interactions via five computer simulations. The first two simulations represented the most ideal circumstances, where the model used to generate the data matched onto the data generating procedure. The last three simulations were designed to violate the distributional assumptions of the covariate model in the model-based procedure.

## 5.1 Summary of Findings

The computer simulations overall suggest that model-based imputation procedure is quite effective when applied to two-level models with a cross-level interaction and random slope. Simulation 1 investigated the performance of model-based imputation under the most ideal circumstance where predictors are multivariate normal. Model-based imputation was largely unbiased, even with 30% missing data rate on both covariates. This finding matches onto previous simulations in the single-level literature (Bartlett et al., 2012, 2014; Kim et al., 2018, 2015; Zhang & Wang, 2017) and multilevel literature (Enders et al., in press; Erler et al., 2019, 2016). In comparison, using the alternative JAV imputation strategy resulted in large amounts of bias, even at high sample sizes. This is also in line with previous single-level research (Enders et al., 2014; Seaman et al., 2012). For sample size considerations, Simulation 1 results suggested that having at least 50 level-2 clusters and 25 level-1 units is sufficient to provide approximately unbiased estimates under a high missing data rate. These sample sizes could be reduced to as low as 25 level-2 clusters in the 15% missing data rate condition. Therefore, one would expect that model-based imputation will provide near-optimal performance when using the ‘30/30’ rule of thumb sample size that is often suggested (Hox, Moerbeek, & Van de Schoot, 2017; Kreft & de Leeuw, 1998; Snijders & Bosker, 2012) Nevertheless, even in the smaller sample size conditions, model-based imputation outperformed the alternative JAV approach.

Simulation 2 was a modification to the first simulation where a level-2 covariate was binary. Similar to the findings of Simulation 1, model-based imputation outperformed JAV across the board. Generally, model-based imputation had acceptable estimates with sample sizes as low as 50 level-2 clusters and 25 level-1 units with a 50/50 split between categories; however, this sample size recommendation was dependent on other design factors. For example, in the 80/20 condition, the simulation suggests that much larger sample sizes are needed (e.g.,  $L1 = 25$ ,  $L2 = 200$ ). In part, this is due to how the MAR mechanism was induced, which selected the 20% category as more likely to be missing than the 80% category; thus, a higher sample size is needed at level-2 because the 20% category had even fewer observations

due to the missing data (e.g., with 50 clusters, on average there was 5 to 6 observations in the 20% category). It is important to note that changing the deletion mechanism to remove scores from the 80% category would dramatically change the results, and the model-based method would most likely perform even better. Despite these shortcomings, model-based imputation still outperformed the alternative approaches of listwise deletion or JAV.

Both Simulations 3 and 4 investigated forms of between-cluster non-normality. Simulation 3 considered the case where a level-2 variable was non-normally distributed, and Simulation 4 investigated the case where the group means of a level-1 variable were non-normally distributed. Results from both simulations suggested that there was a small increase in bias of a few percent compared to Simulation 1. The misspecification in Simulation 3 had an effect on both the regression parameter for the level-2 predictor and the cross-level interaction, but these biases fell below 10% in the larger level-2 sample size conditions. Enders et al. (in press) had a similar finding when investigating non-normality of a level-2 variable in a random slope model. Turning to Simulation 4, model-based imputation was robust to the between-cluster non-normality in the level-1 covariate's latent group means. Compared to Simulation 1, Simulation 4 had very similar observed biases. In contrast, Simulation 3 had more observed bias in both the level-2 covariate and the cross-level interaction. Despite these findings, caution is still warranted. As discussed below, the Simulation 5 results were quite sensitive to the interaction between distribution shape and missing data mechanism. Further study is needed to determine whether the findings in Simulation 3 and 4 generalize to other settings.

Simulation 5 investigated misspecification of the within cluster residual of the level-1 predictor. The goal of this simulation was to induce highly non-normal data within each cluster. The findings suggested that both model-based imputation and JAV performed poorly with this type of misspecification. Both methods had bias over 40% for both the regression slope for the cross-level interaction and the random slope variance; however, caution ought to be warranted when interpreting these results. Firstly, as Figure 25 illustrates, the data generation produced a rather pronounced cluster-specific floor effect. While not impossible

with real data, it is difficult to imagine substantive applications where such a cluster specific floor effect would be produced. For example, a measurement process can have an floor effect when a scale is not sensitive enough to or does not include questions that measure very low levels of the construct; however, such a process would yield a common floor effect, not one that varies by cluster. Secondly, as the follow up simulation suggested, the magnitude and direction of the bias was highly dependent on how the MAR missing data mechanism was induced. By altering how the MAR mechanism was generated (i.e., allowing lower values on  $Y$  to be associated with higher missingness on  $X$ , rather than the reverse), the biases were reduced to be under approximately 15%.

## 5.2 Limitations and Future Research

The simulation studies presented in this dissertation had several limiting factors that potentially limit their generalizability. Firstly, I only investigated an analysis model with one predictor at level-1, one predictor at level-2, and a cross-level interaction between the predictors. This could be expanded to other multilevel interaction effects (e.g., two level-1 variables), nonlinear polynomials (e.g., quadratic models), or even combinations of both. Additionally, I focused on an analysis model where the dependent variable was always complete and normally distributed. These simulations could be expanded to investigate ordinal and nominal outcomes, as the model-based procedure readily accommodates this possibility. Another important limitation was that the cross-level interaction was generated in such a fashion that its coefficient was the same at both level-1 and level-2. The cross-level interaction effect between a group mean centered level-1 variable and the level-2 variable need not be the same as the between-cluster interaction involving the group means and the level-2 variable (Preacher, Zyphur, & Zhang, 2010); thus, it is necessary to develop imputation methods that can correctly accommodate analysis models that seek to partition interaction effects into their constituent parts.

In addition to the limited models investigated, the forms of misspecifications across the simulations were also limited. For example, Simulations 3 to 5 only focused on non-normality

of the between- and within-cluster residuals. While this is one method of inducing non-normality in the data, an alternative procedure could define a level-1 predictor as the sum of a normally distributed cluster mean and within-cluster residual, then transform the full variable to some desired distributional shape. For example, had I used this approach with Simulation 5, the resulting data would still be substantially non-normal, but the entire sample would share the same floor effect. The simulation studies were also limited by the functional form of how the MAR mechanism. For example, my procedure associated higher values of the outcome with missingness on the covariate, but future simulations could investigate situations where the center of the distribution or the tails were more likely to be deleted. As discussed already, it appears that the imputation method's sensitivity to misspecification is dependent on how the MAR data are generated. In addition to examining distributional misspecifications, future studies should explore different functional forms among the covariates. As a by-product of assuming normality of the covariates (or their underlying latent scores), the version of model-based imputation examined here also assumes linear relations among the covariates. If  $X$  is a quadratic function of  $Z$ , for example, I would expect the procedure to introduce bias. The so-called sequential algorithm (e.g., Erler et al., 2019, 2016) can accommodate certain patterns of non-linearities among predictors, and minor changes to the Bayesian estimation procedure in Blimp could extend the sequential procedure to multilevel models in a very general way

Finally, as with all simulation studies, the generalizability of the results is limited to the conditions that were investigated. Further investigation is warranted in the use of different effect sizes. This includes both the strength of the relationships in the fixed effects and the random effects (i.e., random slope variance). Rights and Sterba (2018) present variance-explained effect sizes for the fixed effects at each level and the random slopes, so future studies could use these new measures to guide the specification of simulation model parameters. Other potential simulation conditions could be the use of different prior distributions (Chung, Gelman, Rabe-Hesketh, Liu, & Dorie, 2015; Gelman, 2006; Gelman et al., 2013; Liu, Zhang, & Grimm, 2016), different ICC values, and additional sample size conditions. For example,

looking at sample size conditions, future studies could investigate the performance of within cluster sizes between 5 and 25. The simulation results suggested that the sample size required to achieve approximate unbiasedness may have been somewhere between the levels that I investigated here (e.g., having within cluster-sizes between 15 to 20, having between-cluster sizes between 50 and 100). Additionally, a future simulation could explore unbalanced cluster size. I have conducted numerous simulations to test Blimp (Keller & Enders, 2019), and it appears that the model-based procedure can handle substantially unbalanced data (as it should theoretically). Nevertheless, this is an avenue for future research.

### **5.3 Conclusion**

In sum, this dissertation outlined and investigated the extension of model-based imputation to multilevel models. This method provided substantial improvements over previous imputation methods when handling cross-level interactions, and researchers can readily implement this method via Blimp (Keller & Enders, 2019) imputation software.

# APPENDIX A

## Figures

• CMP    ⊠ JAV    ◇ MBI

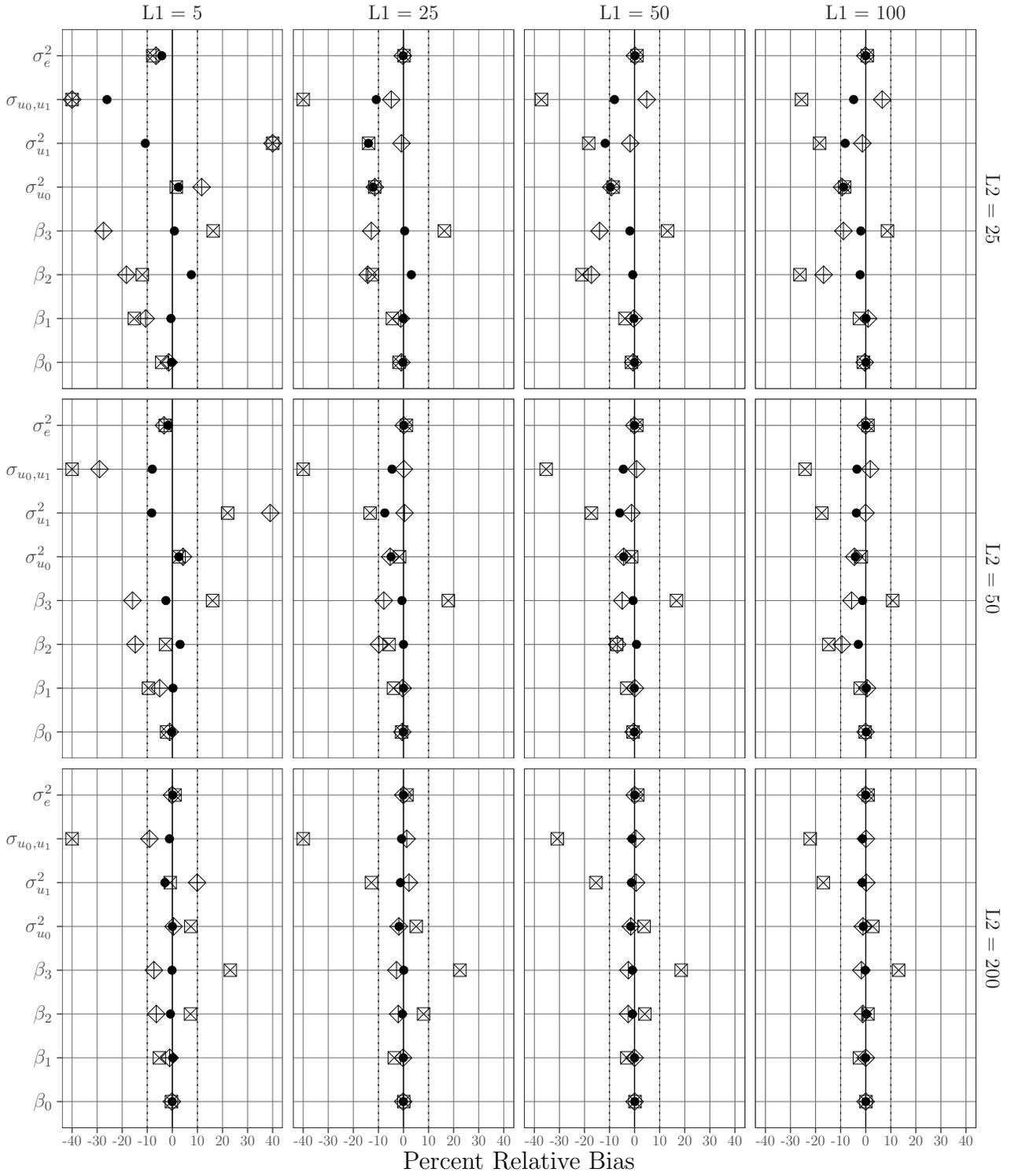


Figure 1: Simulation 1: Percent Bias – ICC = 0.10; Missing = 30%.

● CMP    ⊠ JAV    ◇ MBI

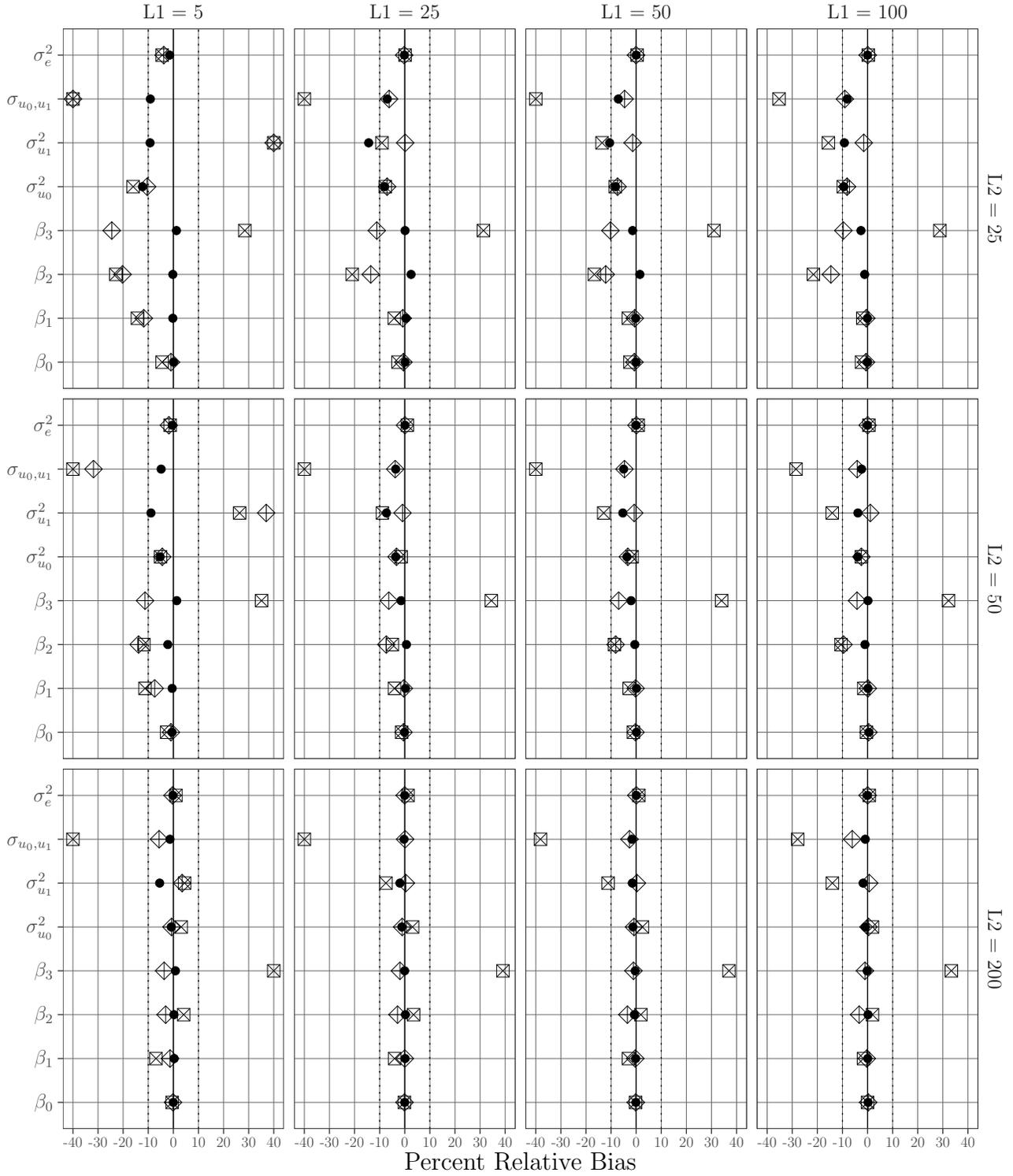


Figure 2: Simulation 1: Percent Bias – ICC = 0.50; Missing = 30%].

• CMP    ⊠ JAV    ◊ MBI

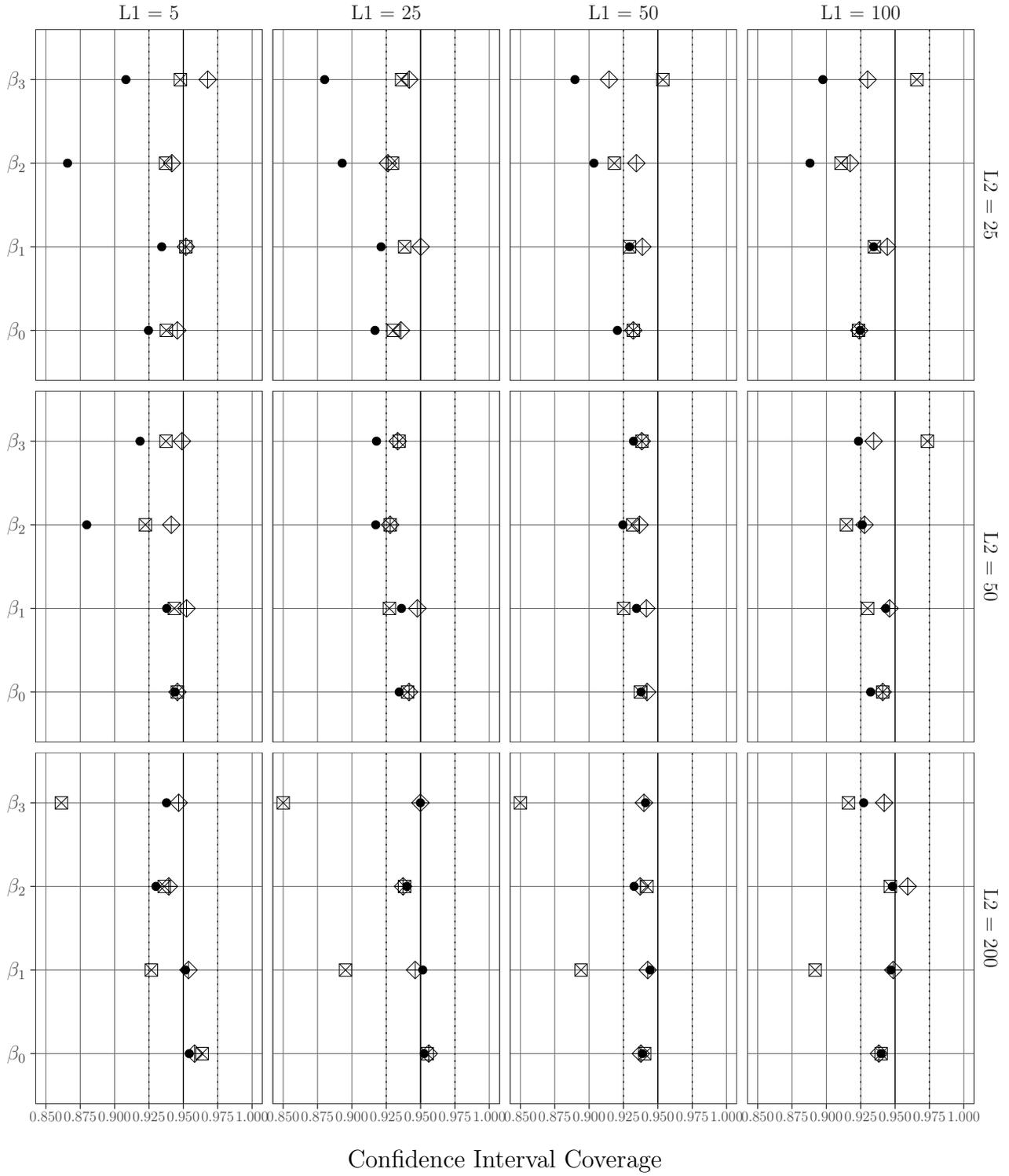


Figure 3: Simulation 1: Coverage – ICC = 0.10; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

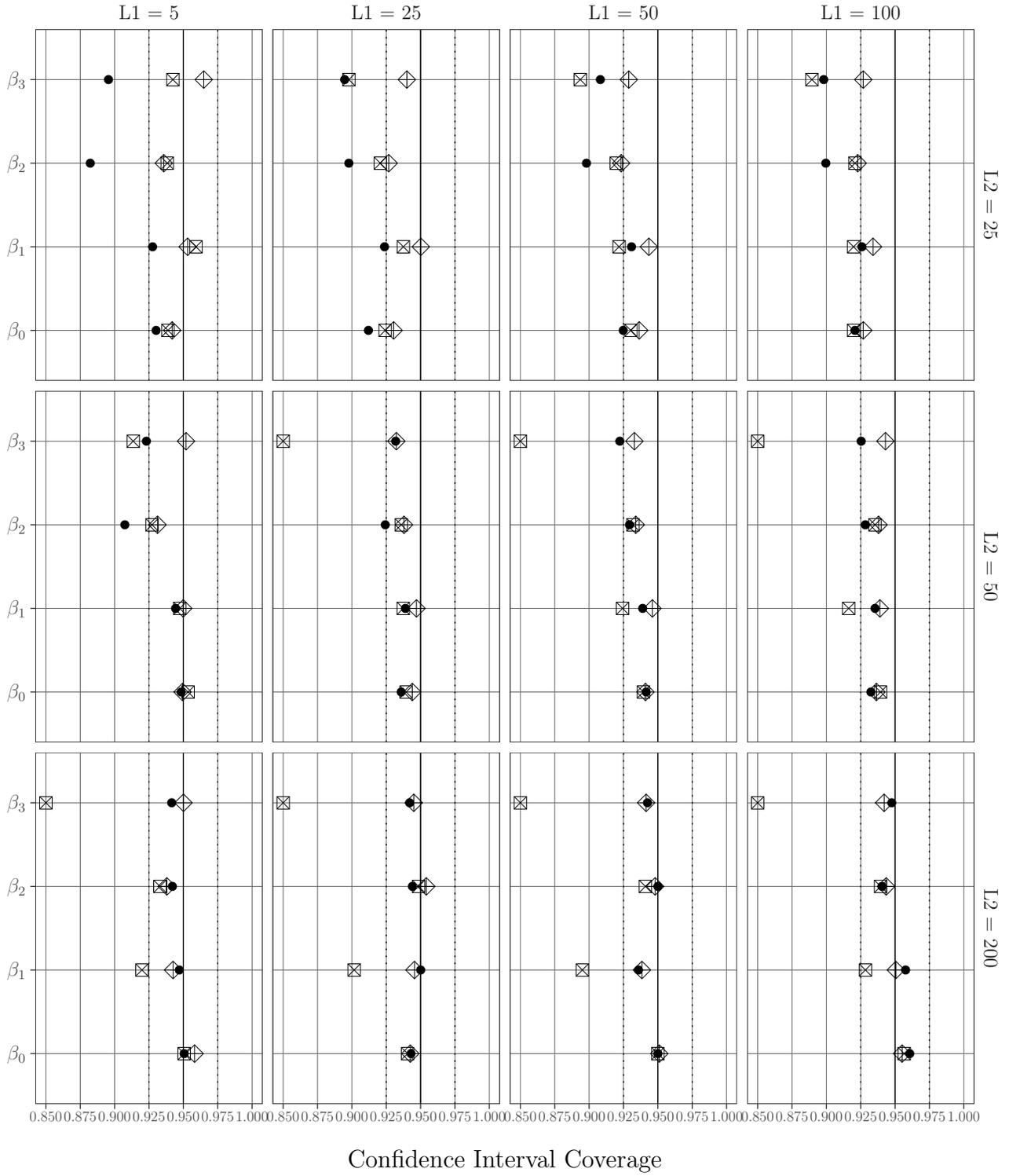


Figure 4: Simulation 1: Coverage – ICC = 0.50; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

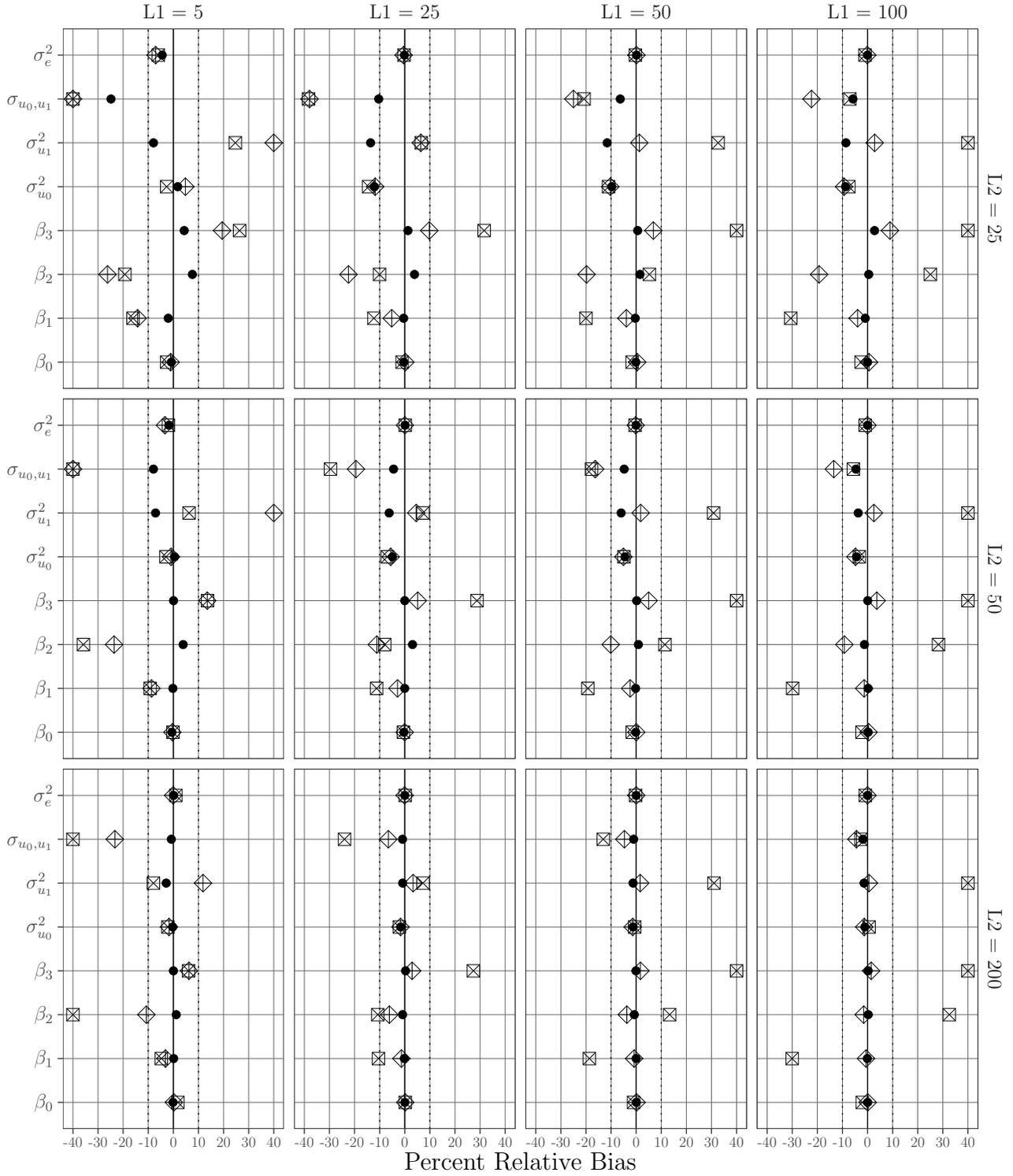


Figure 5: Simulation 2: Percent Bias – ICC = 0.10; Missing = 30%; Split = 50/50.

• CMP    ⊠ JAV    ◇ MBI

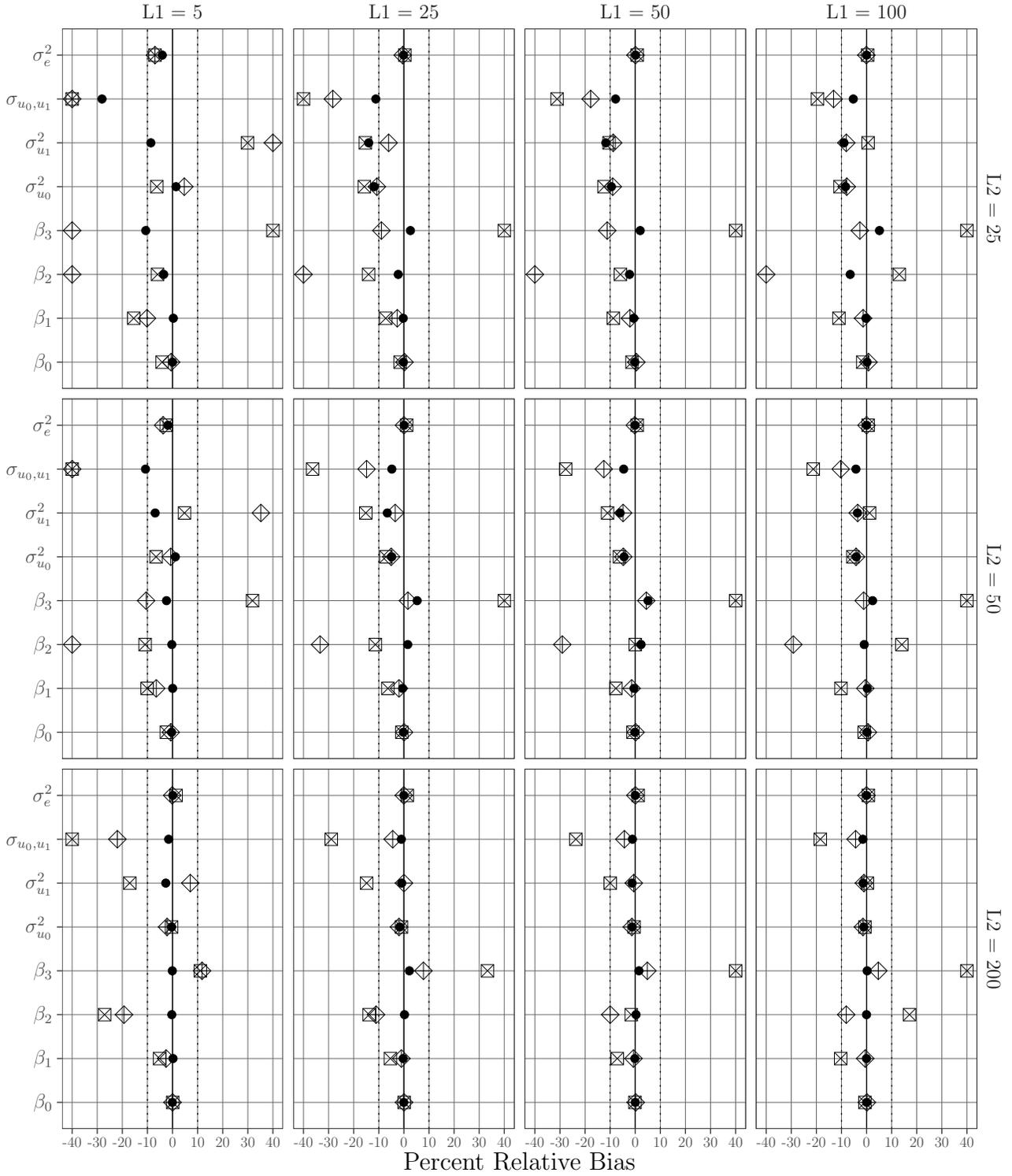


Figure 6: Simulation 2: Percent Bias – ICC = 0.10; Missing = 30%; Split = 80/20.

• CMP    ⊠ JAV    ◇ MBI

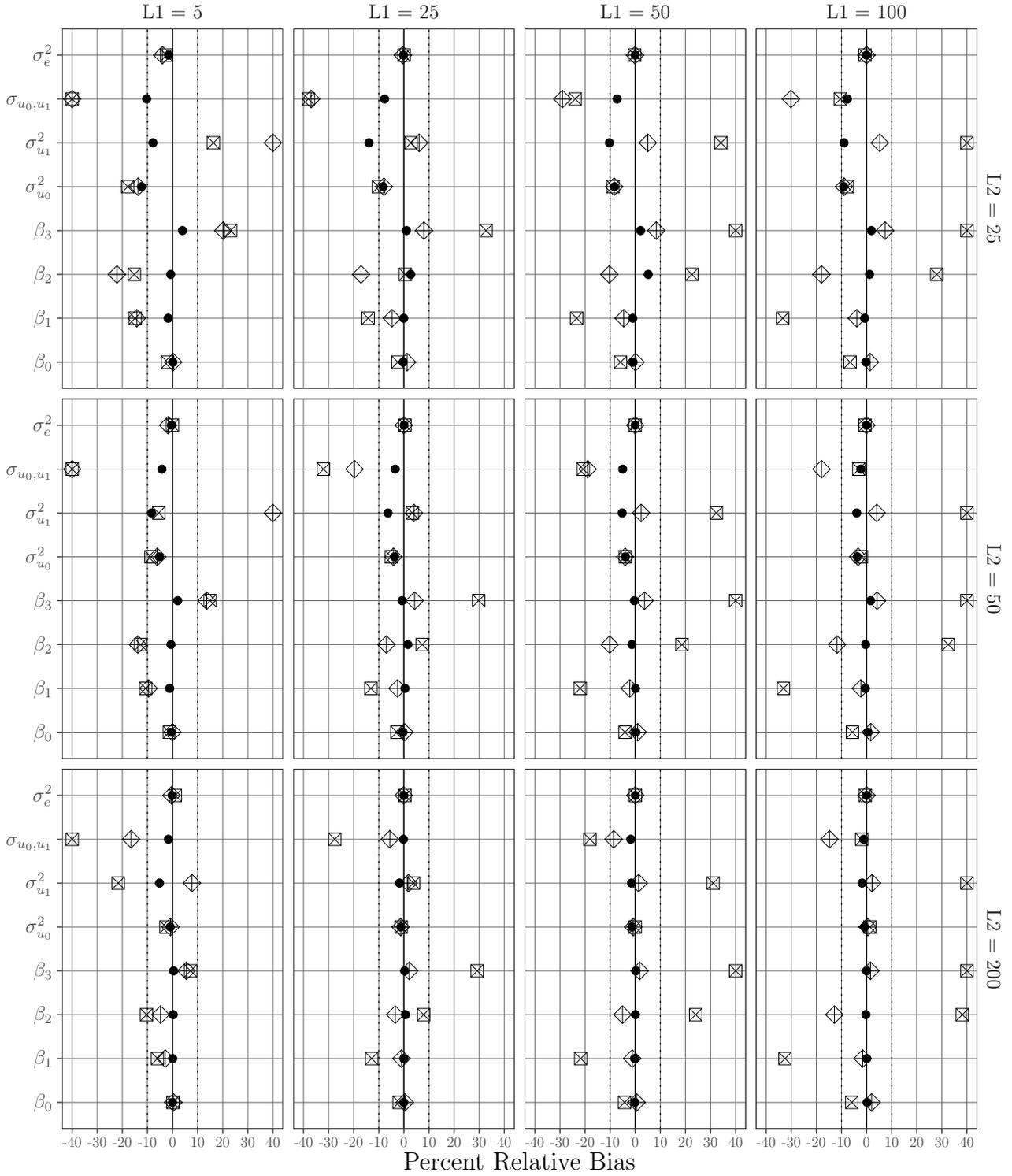


Figure 7: Simulation 2: Percent Bias – ICC = 0.50; Missing = 30%; Split = 50/50.

• CMP    ⊠ JAV    ◇ MBI

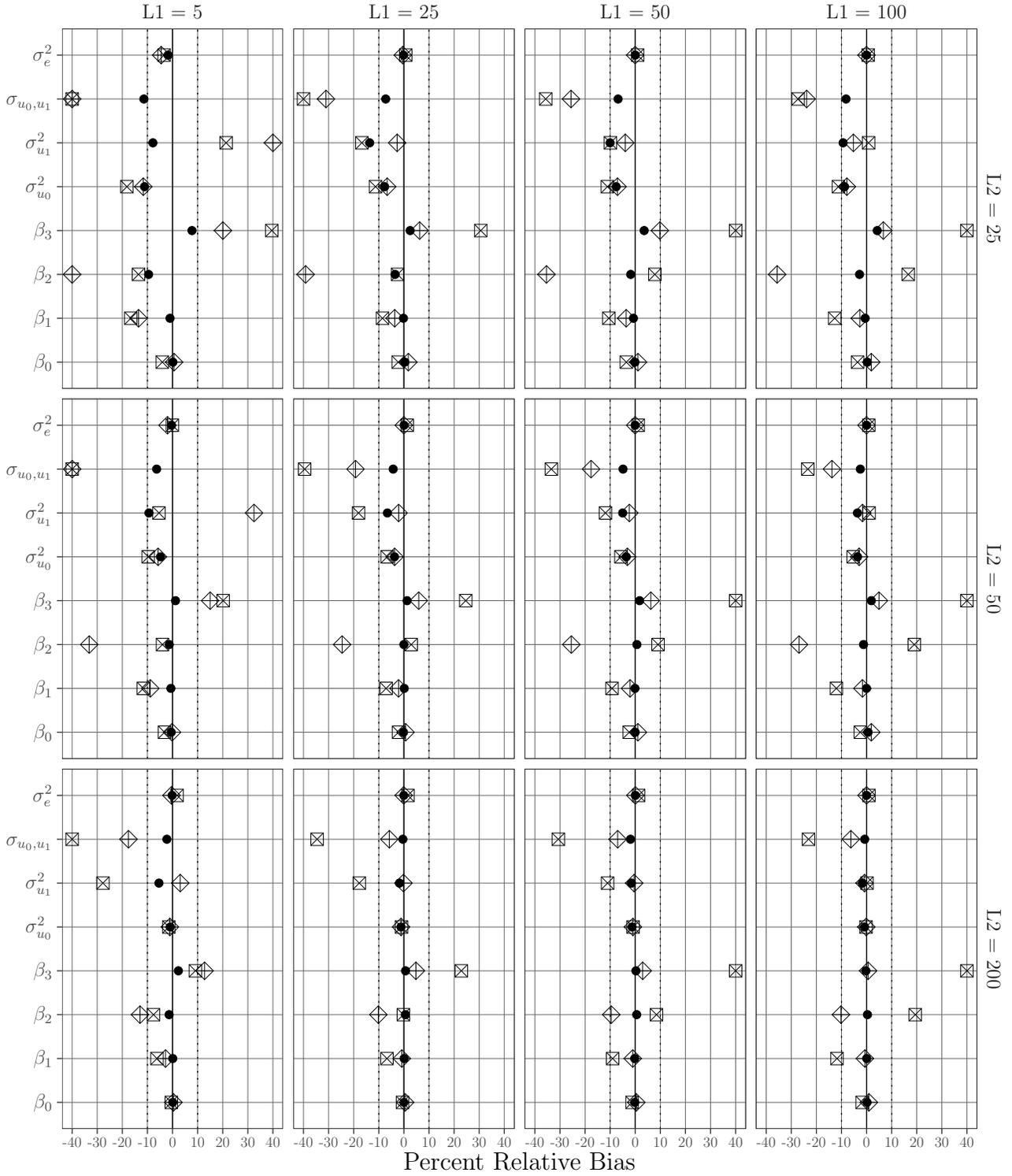


Figure 8: Simulation 2: Percent Bias – ICC = 0.50; Missing = 30%; Split = 80/20.

● CMP □ JAV ◇ MBI

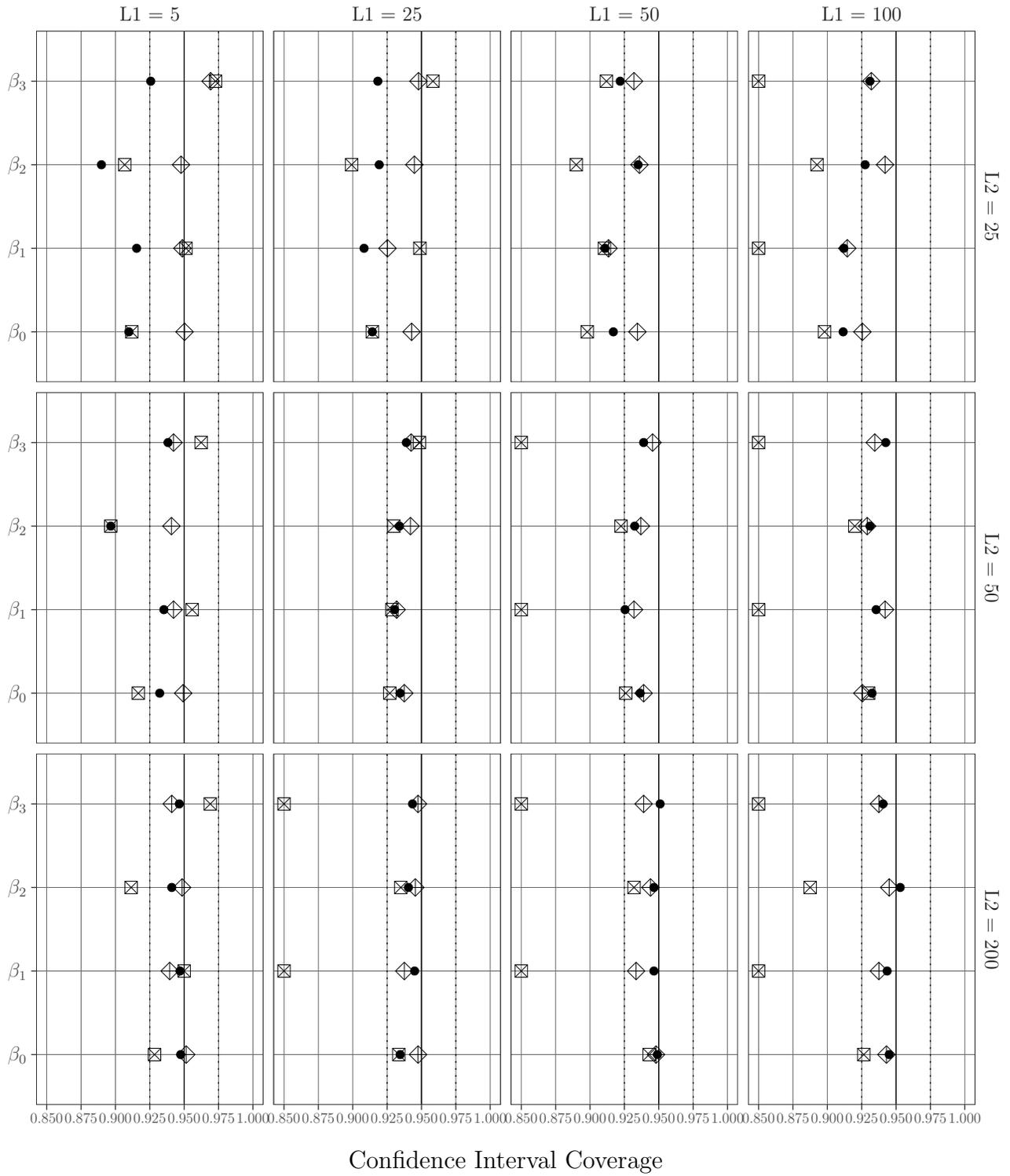


Figure 9: Simulation 2: Coverage – ICC = 0.10; Missing = 30%; Split = 50/50.

• CMP    ⊠ JAV    ◇ MBI

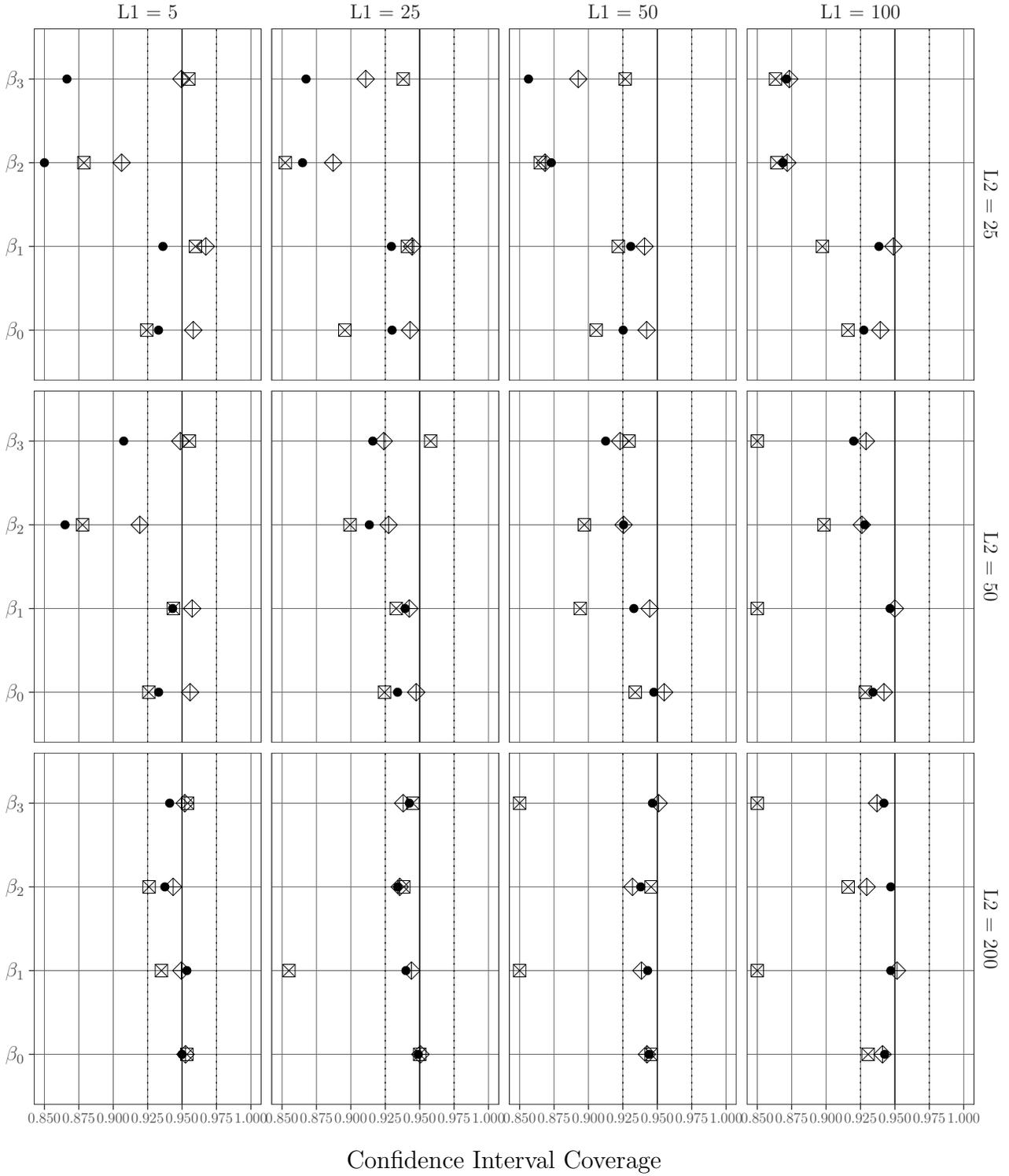


Figure 10: Simulation 2: Coverage – ICC = 0.10; Missing = 30%; Split = 80/20.

• CMP    ⊠ JAV    ◇ MBI

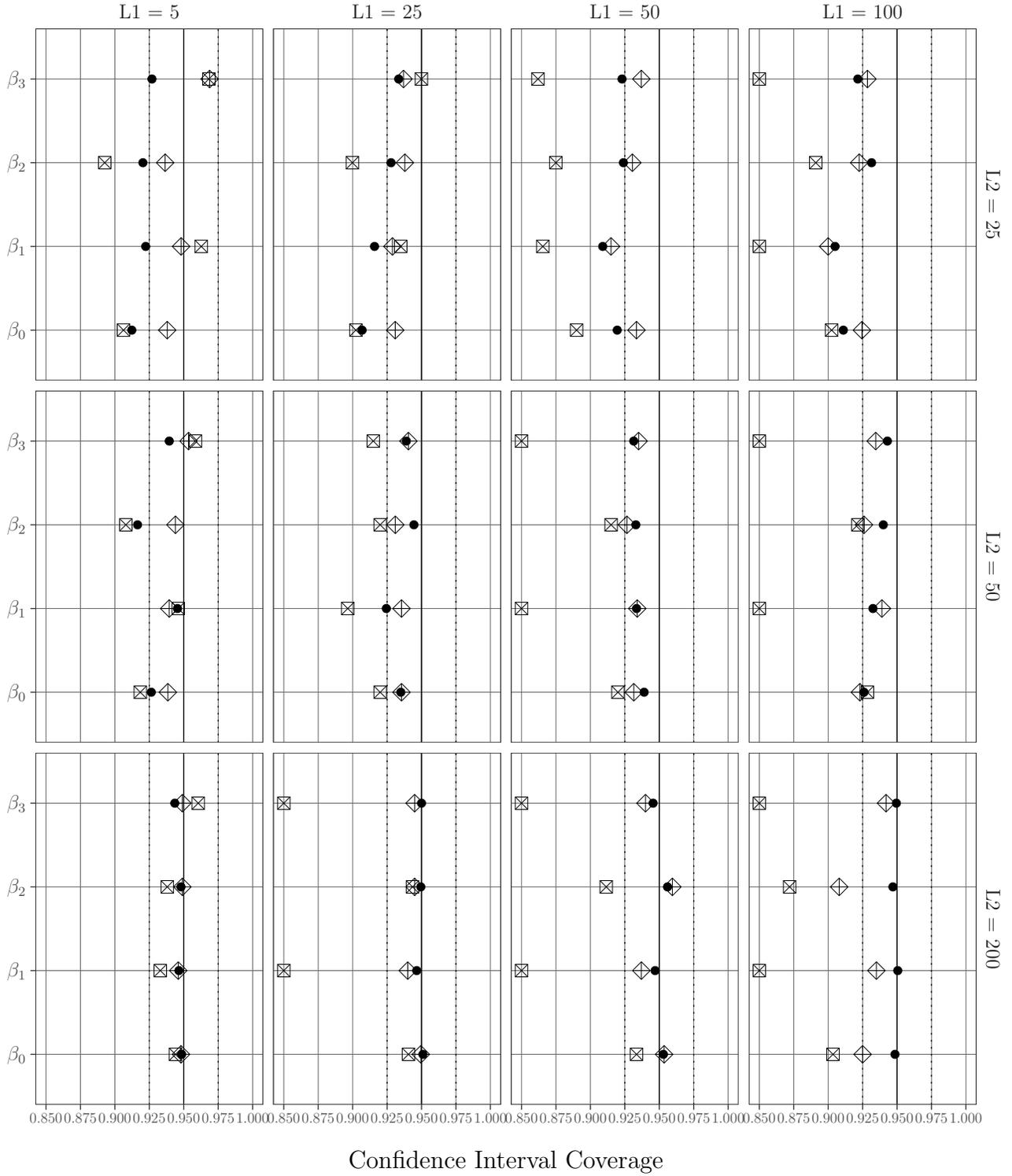


Figure 11: Simulation 2: Coverage – ICC = 0.50; Missing = 30%; Split = 50/50.

● CMP ⊠ JAV ◇ MBI

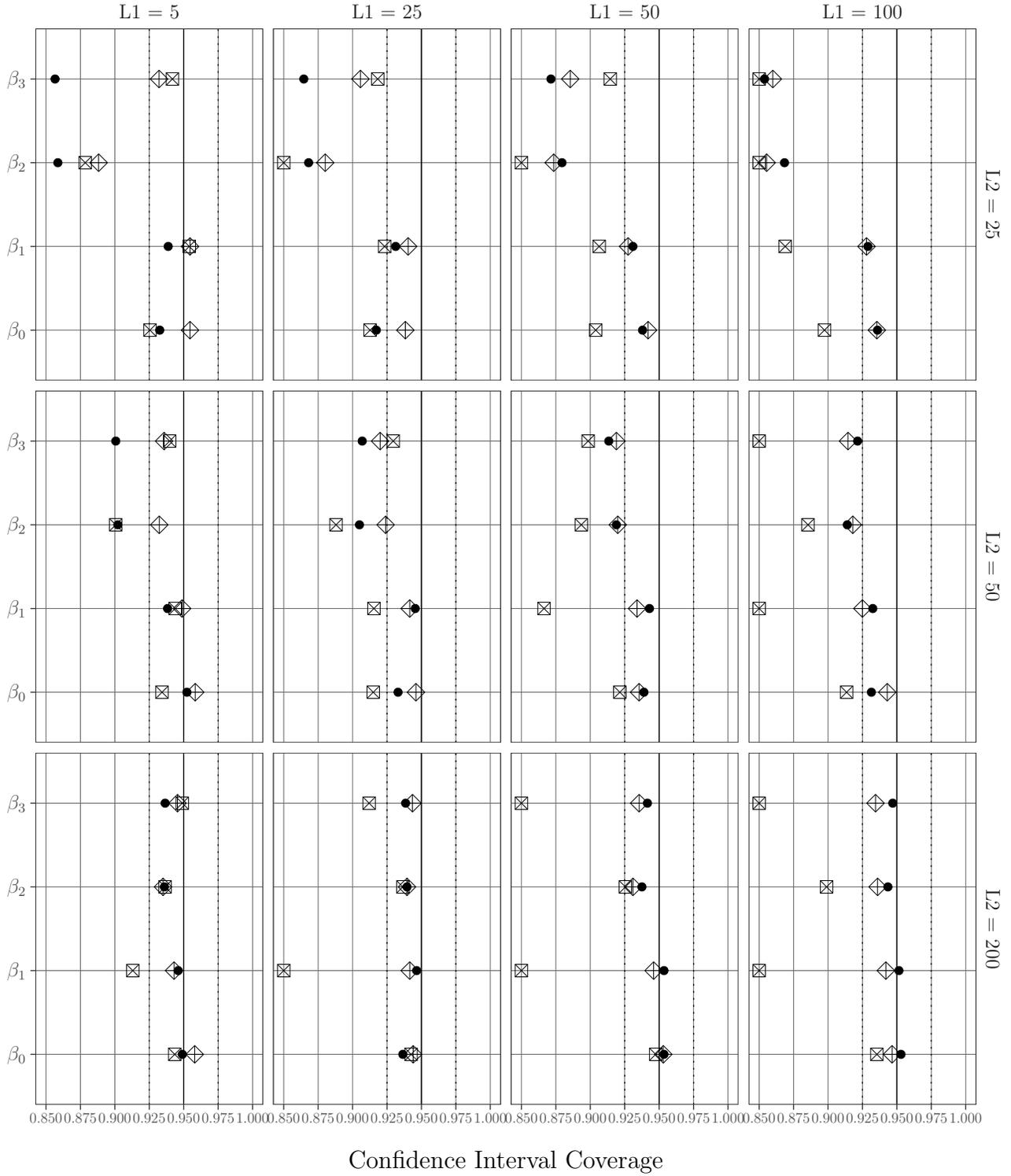


Figure 12: Simulation 2: Coverage – ICC = 0.50; Missing = 30%; Split = 80/20.

• CMP    ⊠ JAV    ◇ MBI

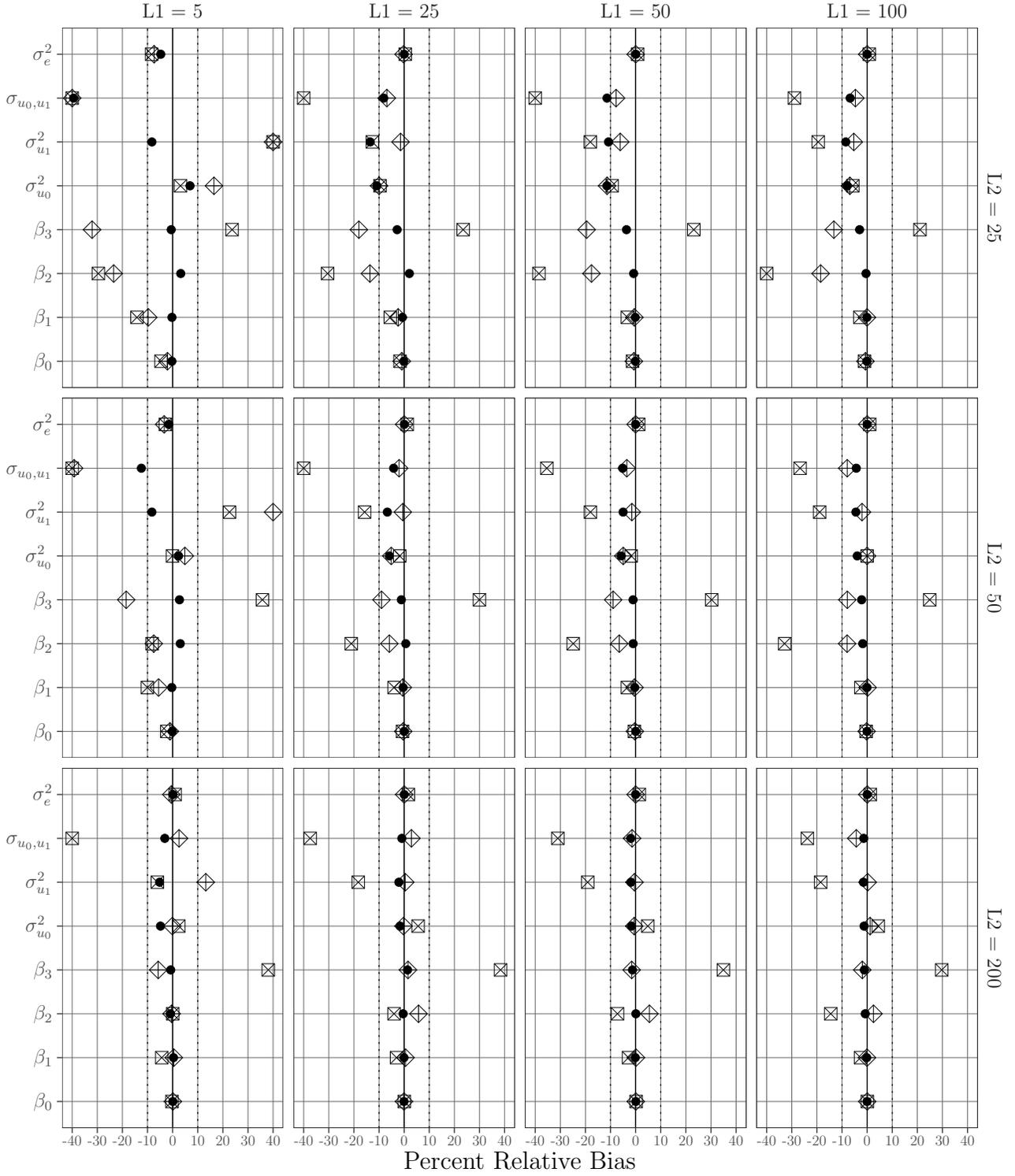


Figure 13: Simulation 3: Percent Bias – ICC = 0.10; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

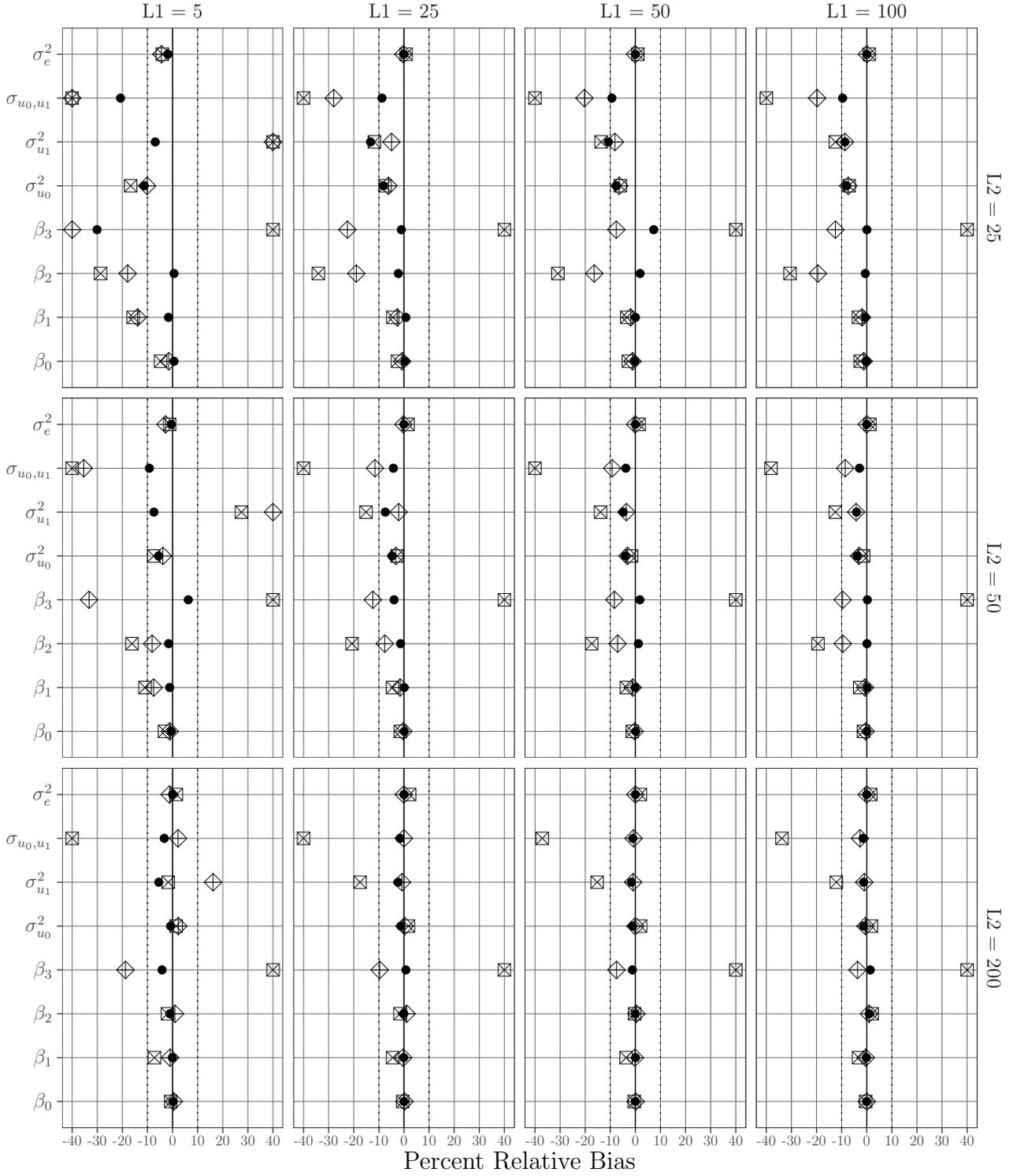


Figure 14: Simulation 3: Percent Bias – ICC = 0.50; Missing = 30%.

● CMP ⊠ JAV ◊ MBI

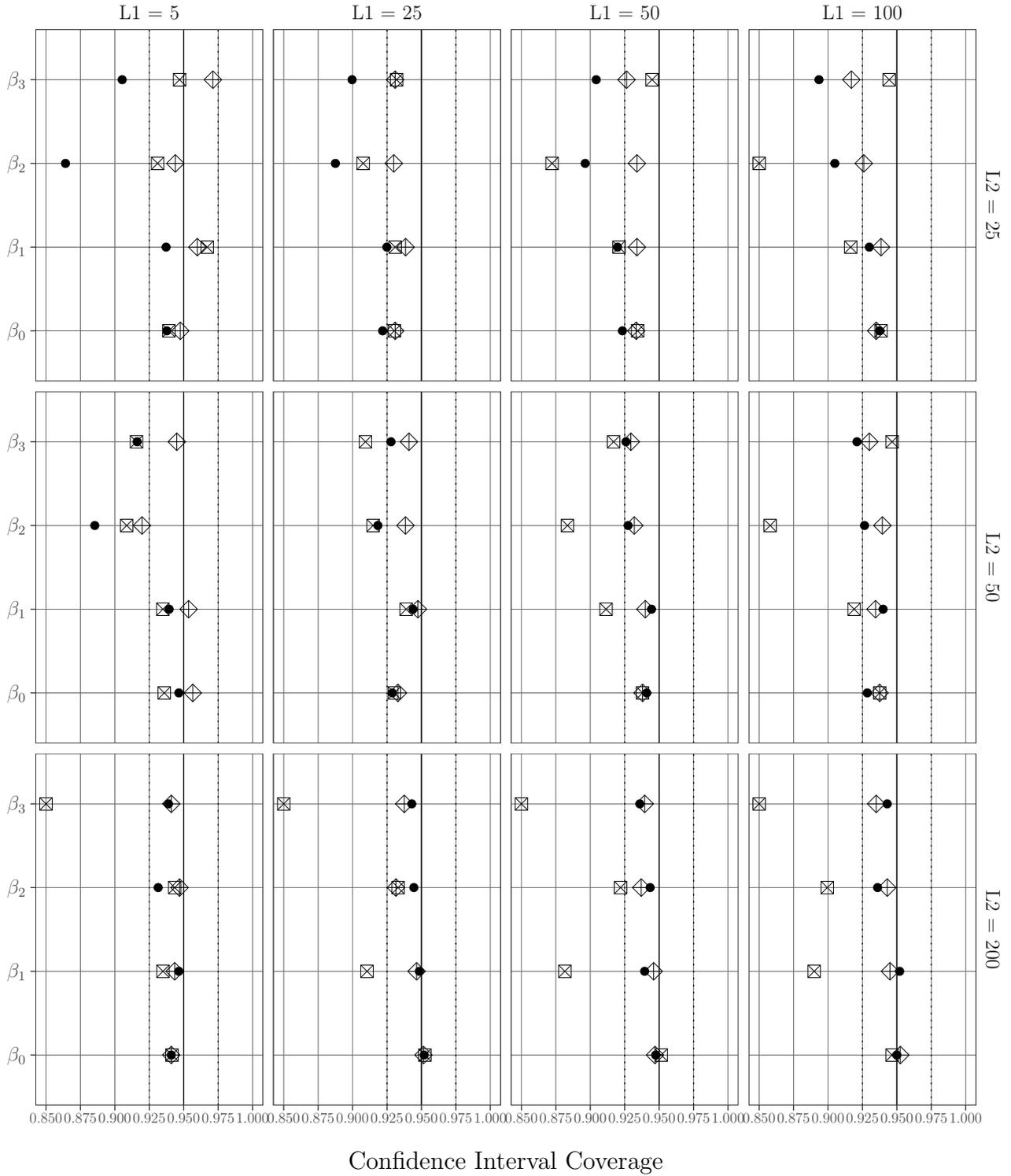


Figure 15: Simulation 3: Percent Bias – ICC = 0.10; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

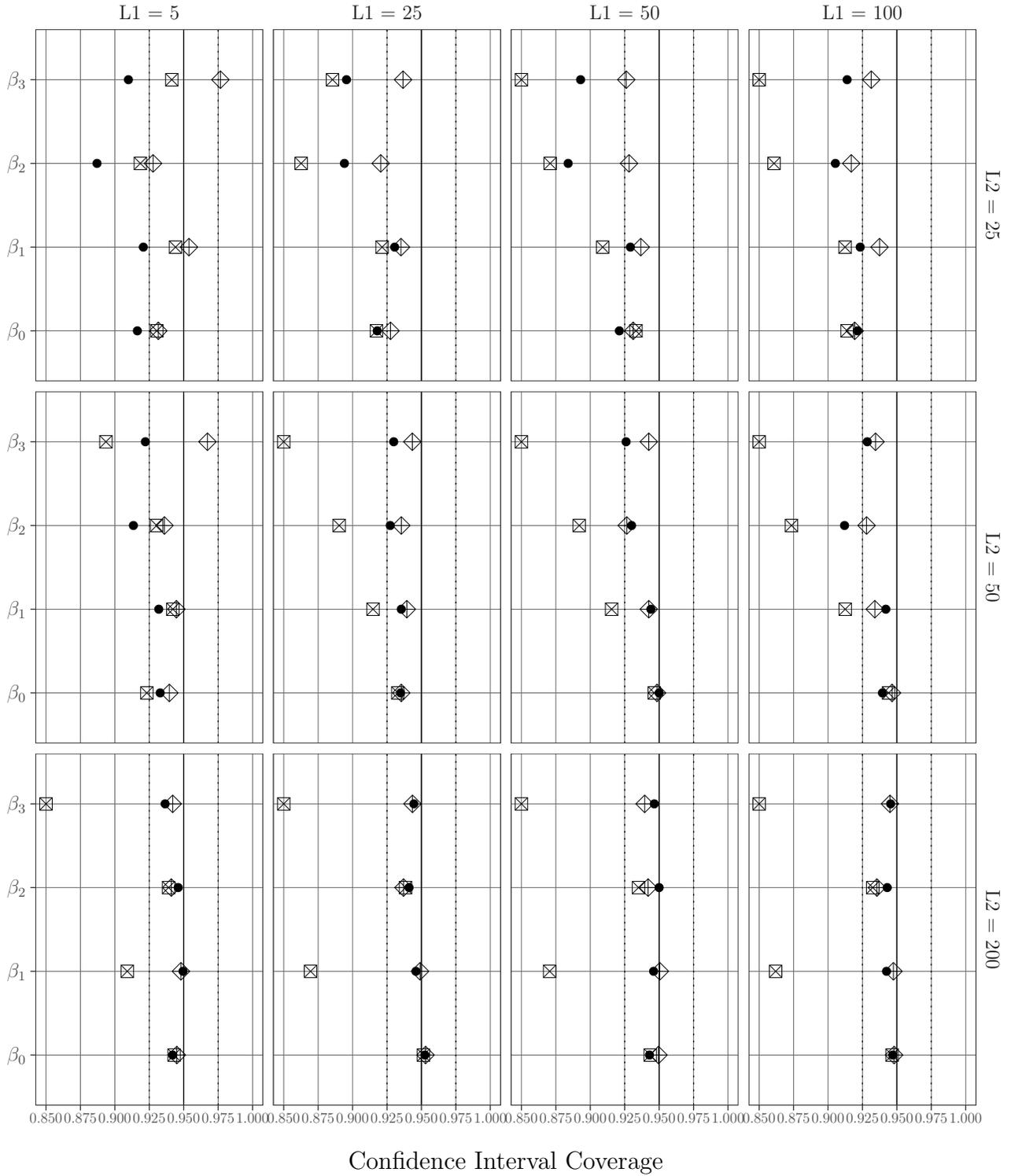


Figure 16: Simulation 3: Percent Bias – ICC = 0.50; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

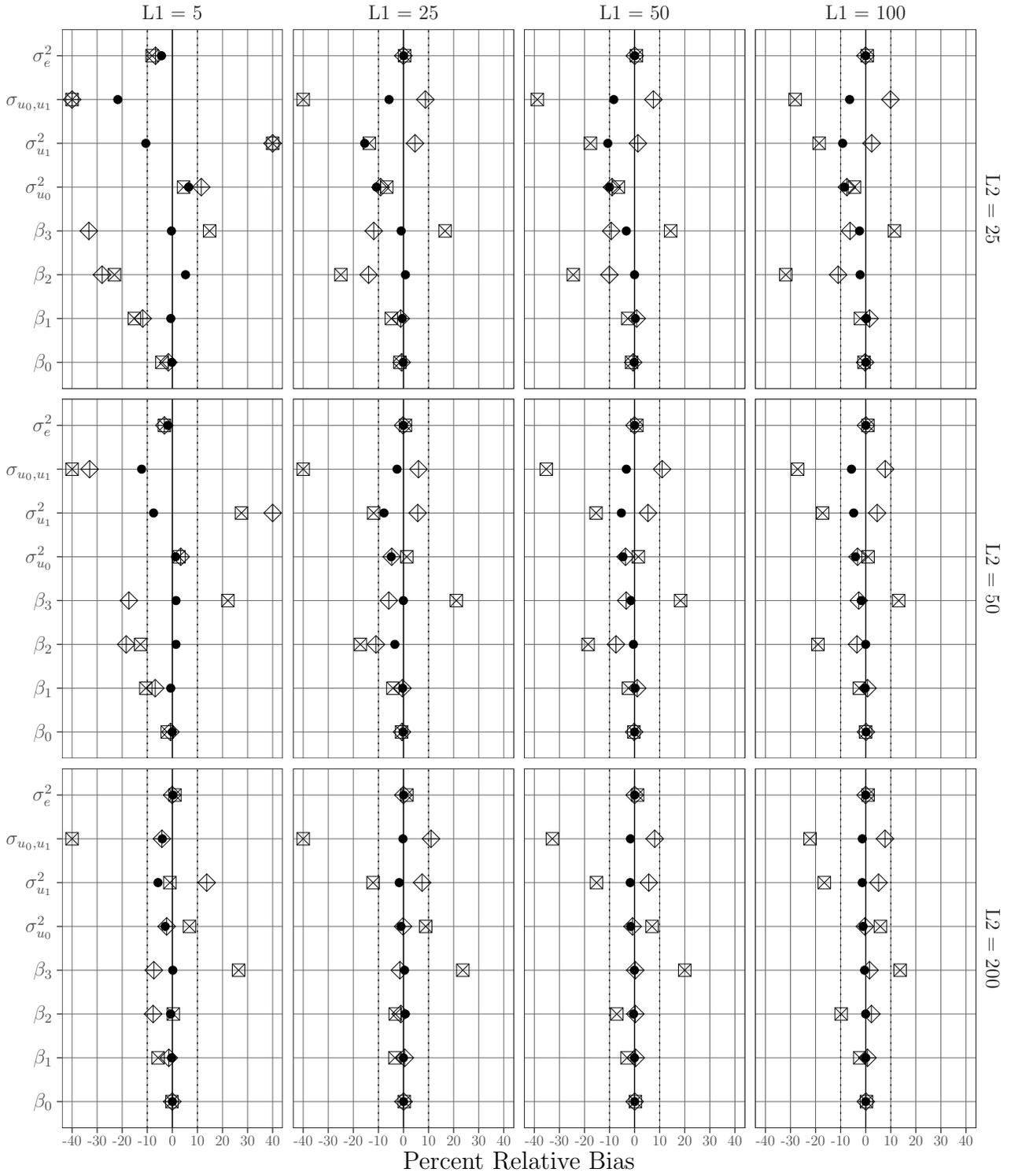


Figure 17: Simulation 4: Percent Bias – ICC = 0.10; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

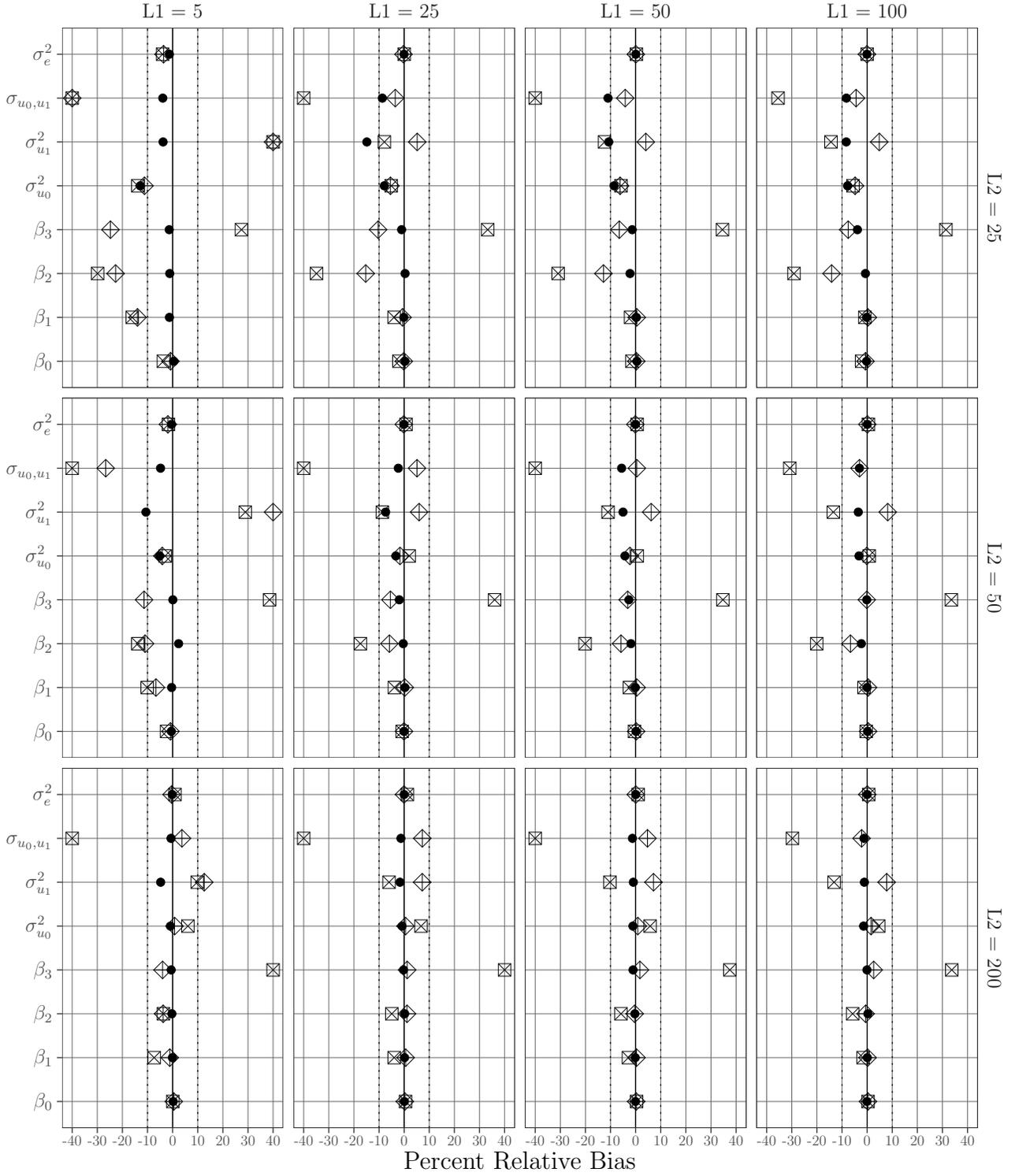


Figure 18: Simulation 4: Percent Bias – ICC = 0.50; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

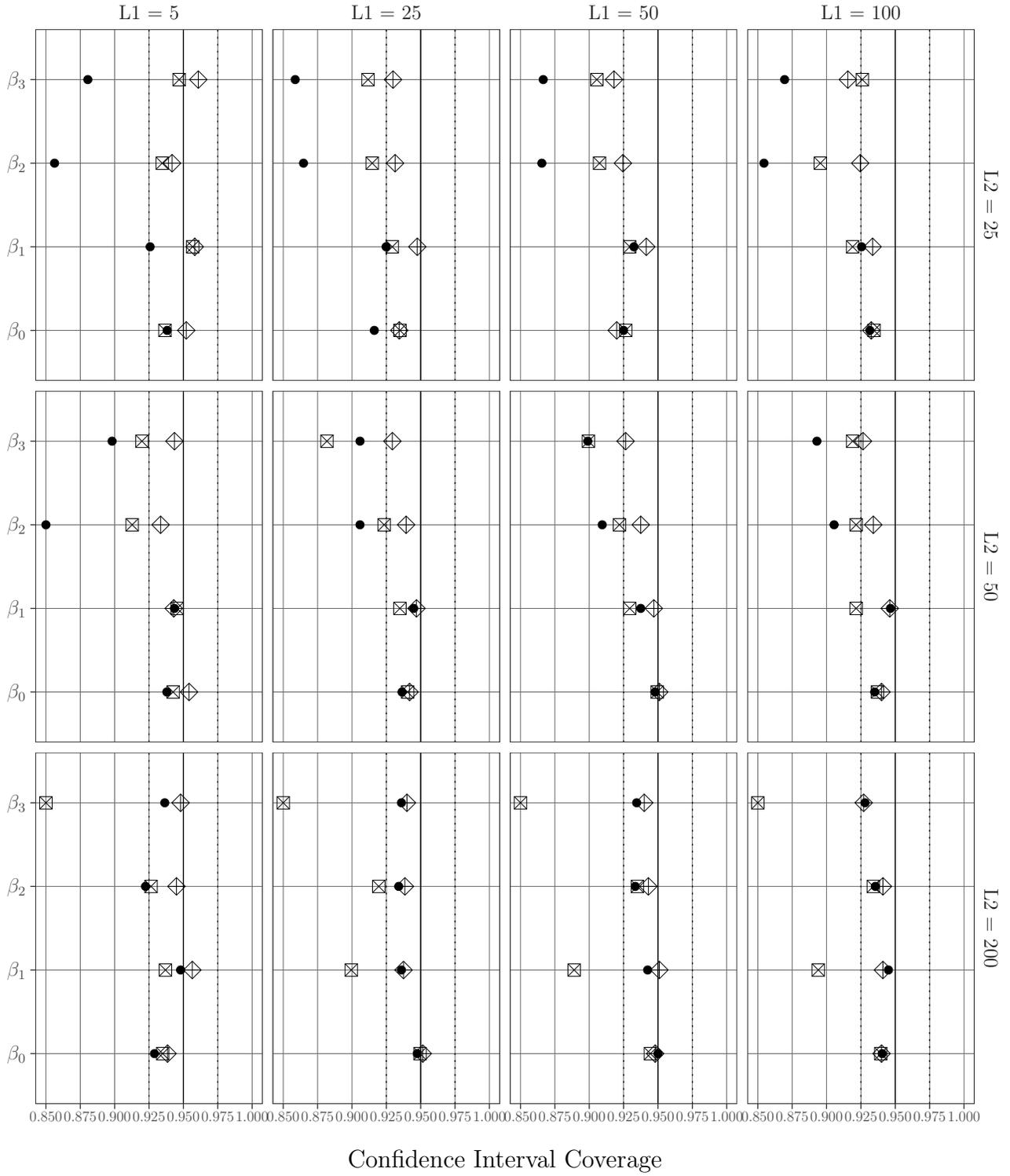


Figure 19: Simulation 4: Coverage – ICC = 0.10; Missing = 30%.

• CMP ⊠ JAV ◊ MBI

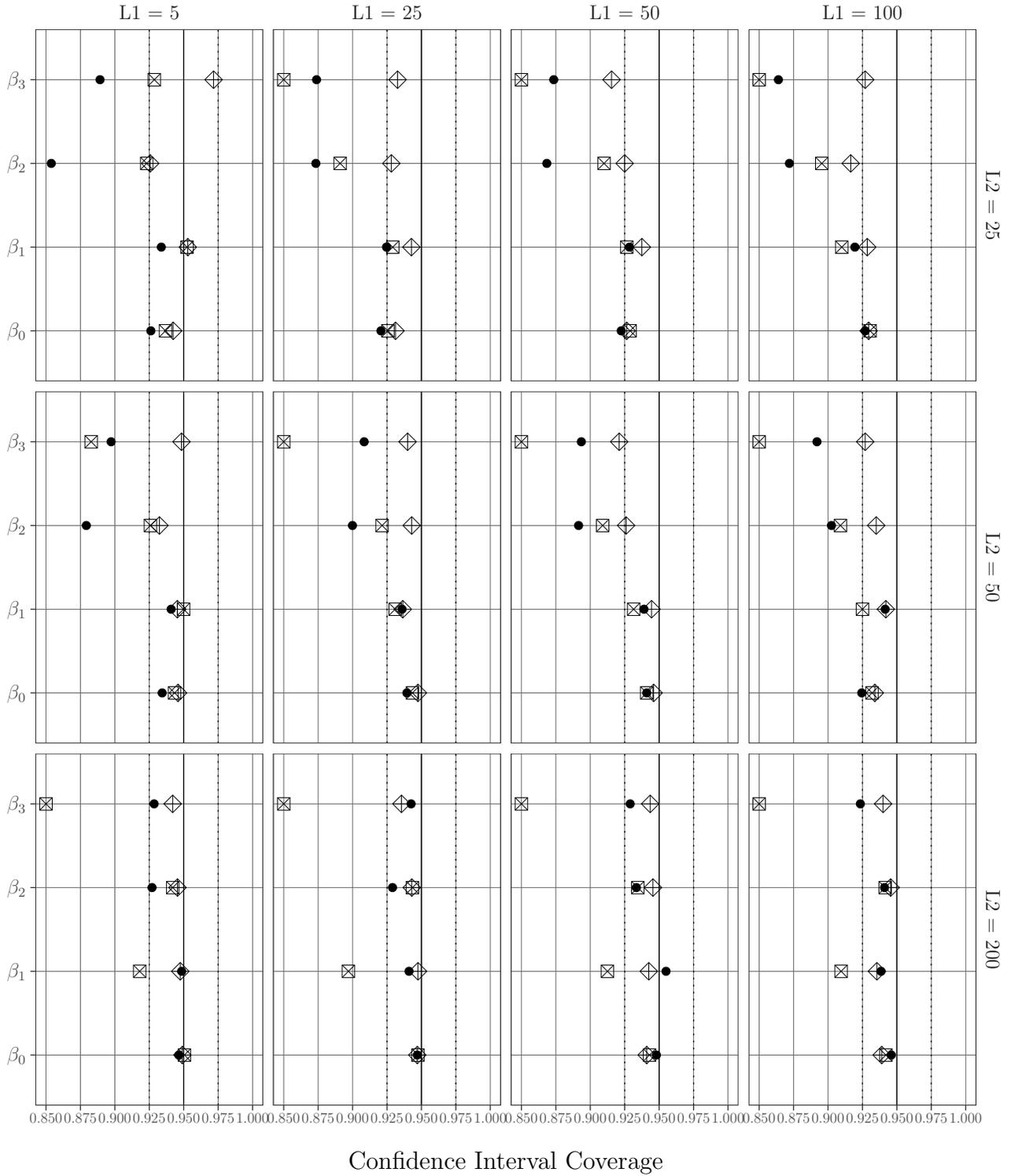


Figure 20: Simulation 4: Coverage – ICC = 0.50; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

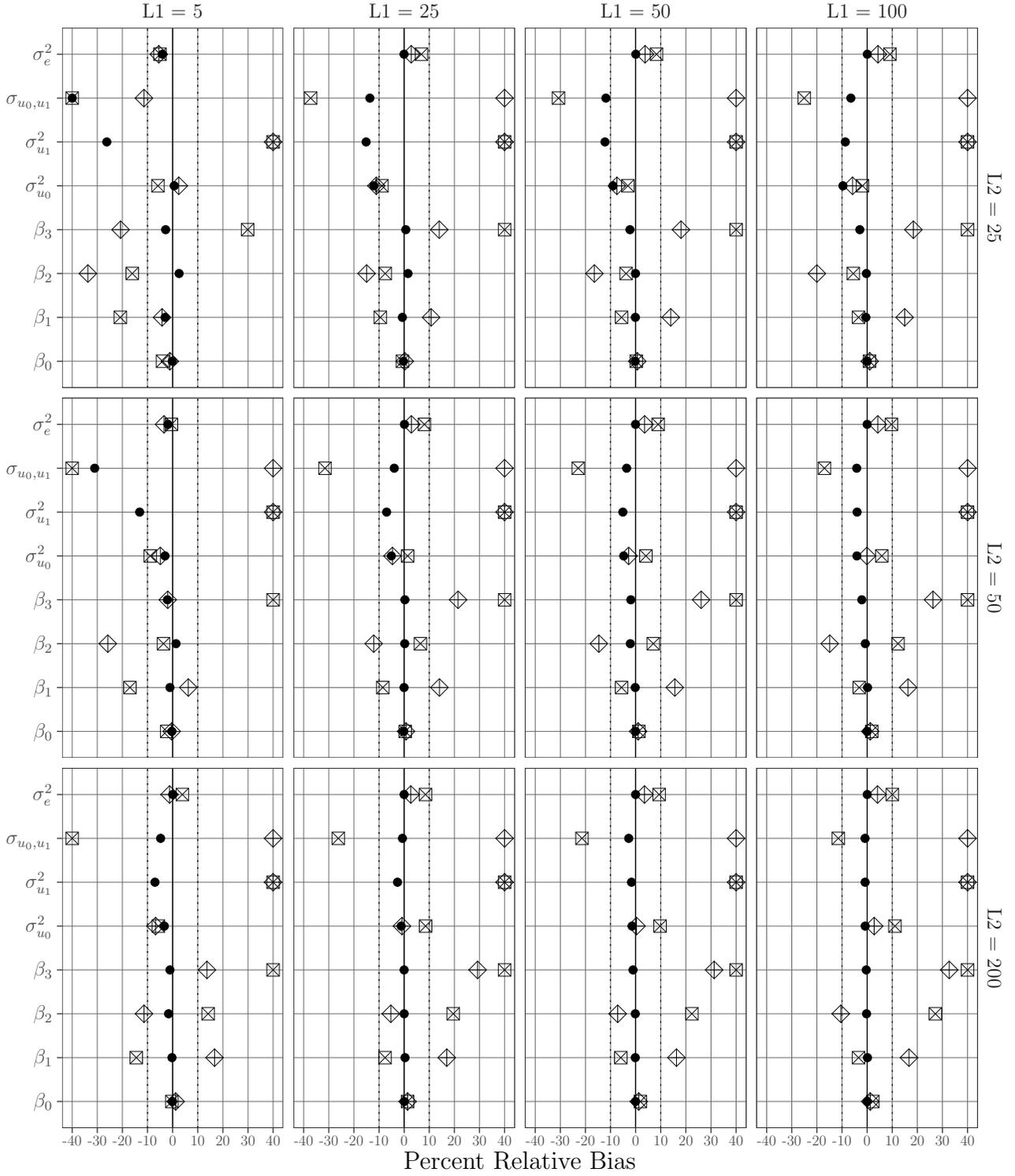


Figure 21: Simulation 5: Percent Bias – ICC = 0.10; Missing = 30%.

• CMP    ⊠ JAV    ◇ MBI

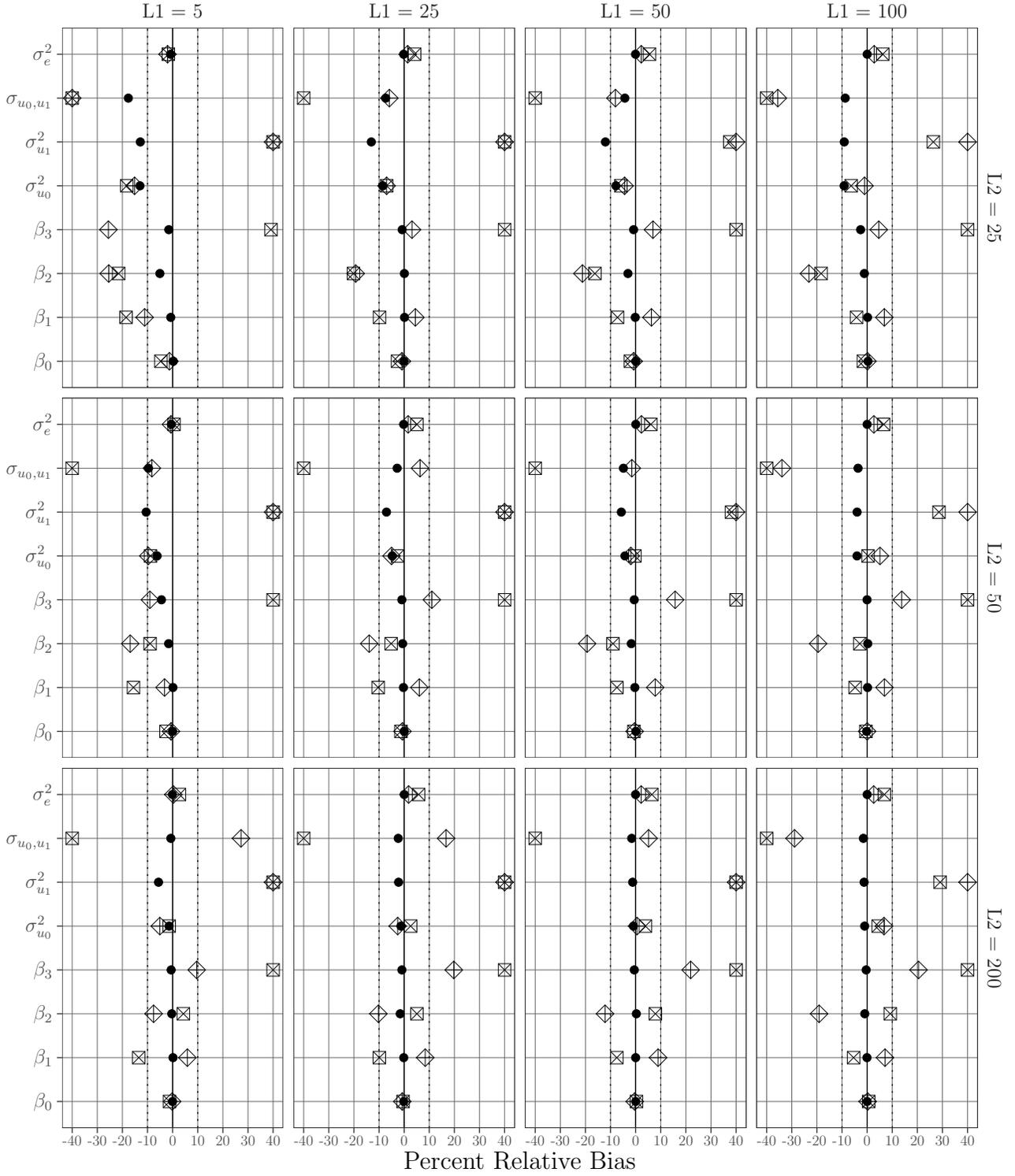


Figure 22: Simulation 5: Percent Bias – ICC = 0.50; Missing = 30%.

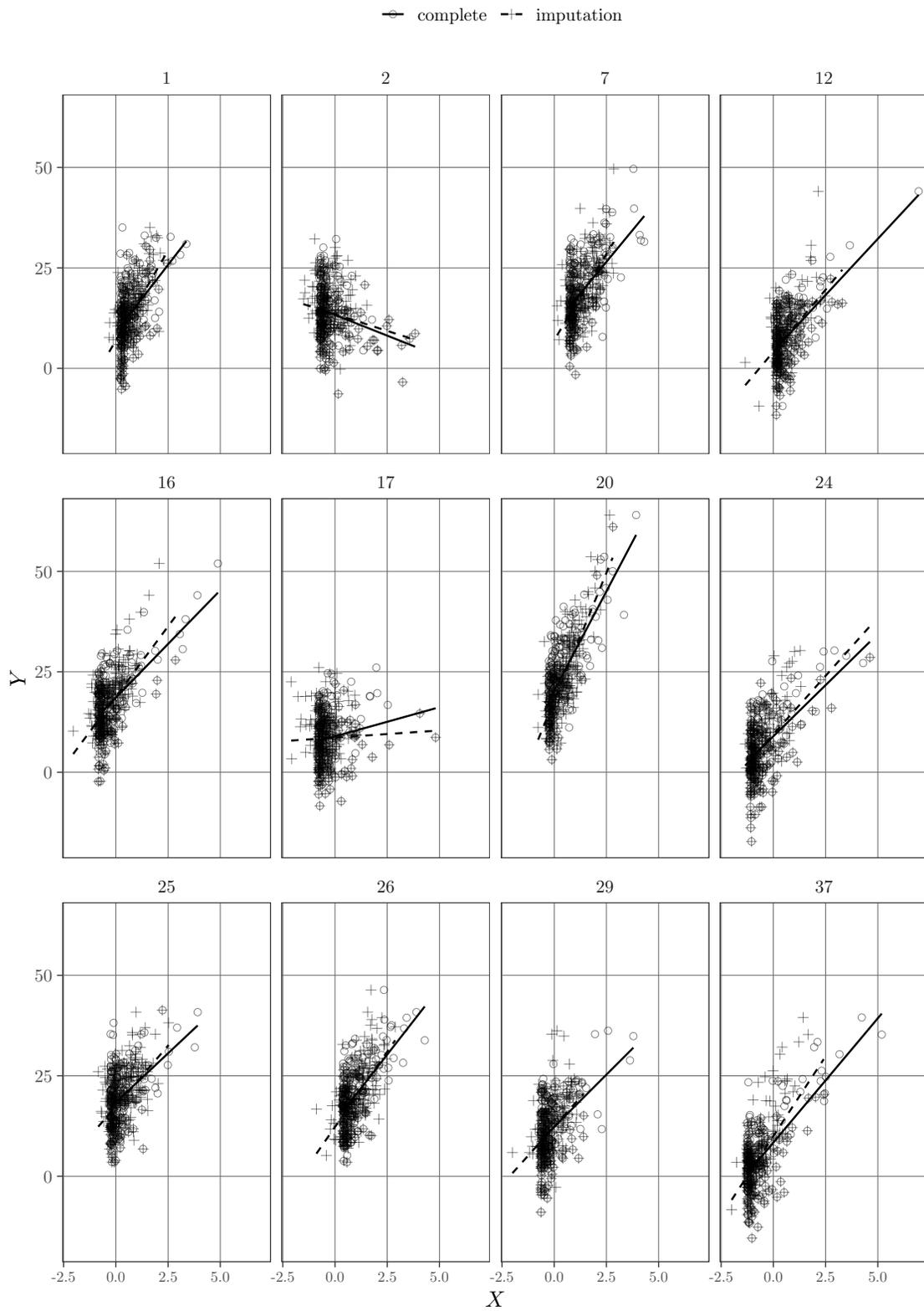


Figure 23: Scatterplot of twelve clusters for  $Y$  regressed on  $X$  for  $\chi^2(1)$  level-1 residual.

• CMP ☒ MBI

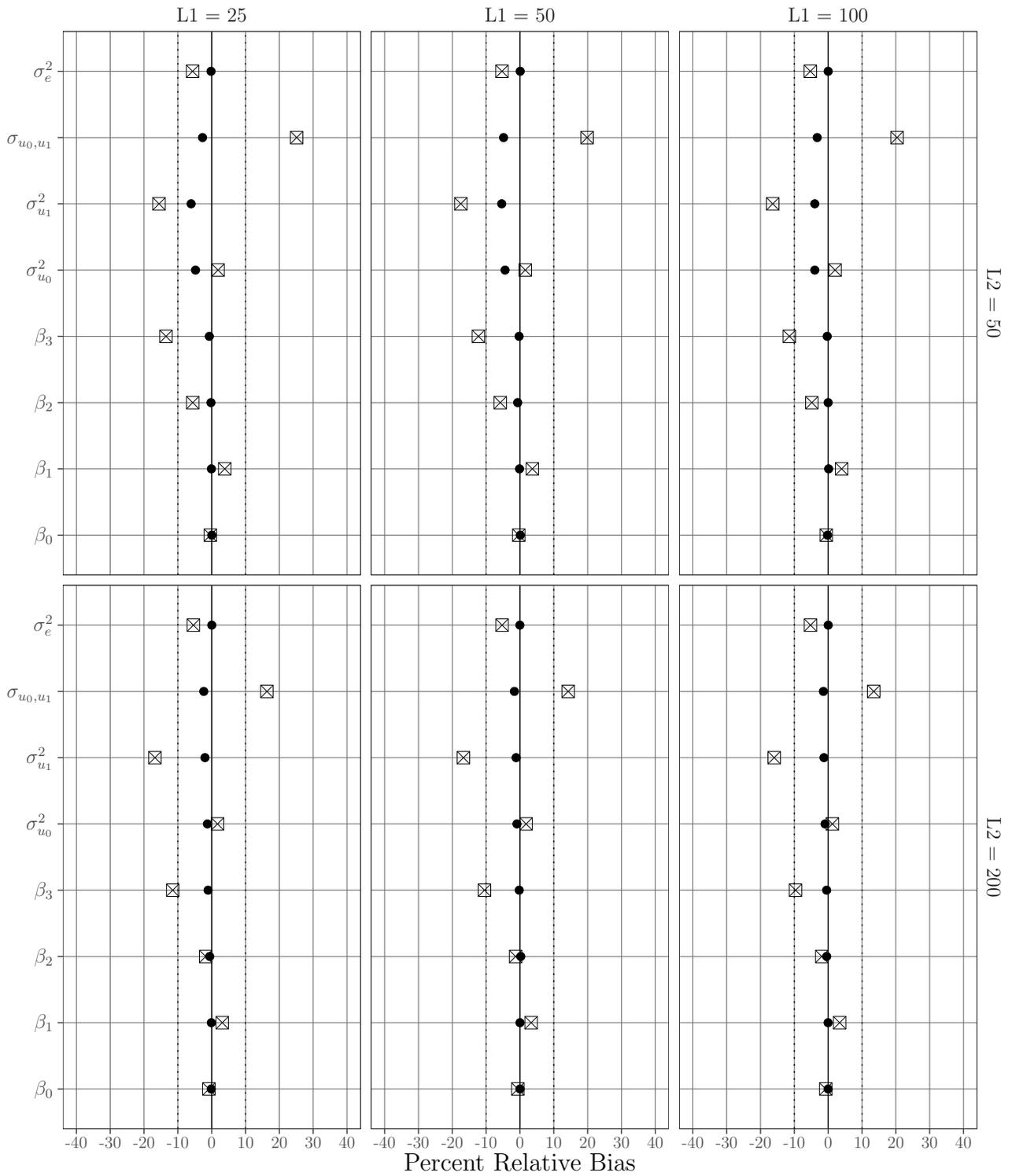


Figure 24: Follow-up Simulation for Simulation 5. Percent Bias – ICC = 0.50; Missing = 30%

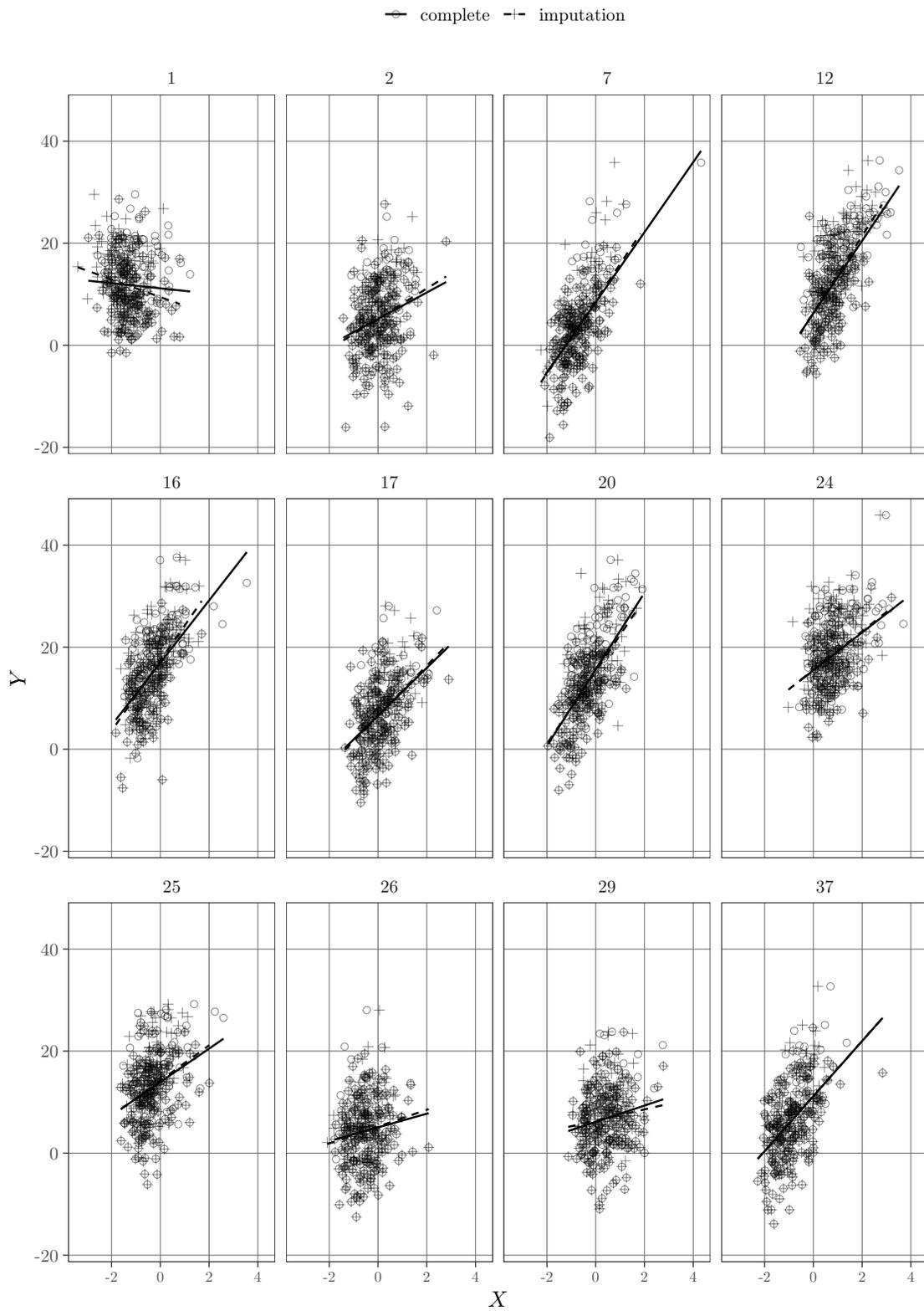


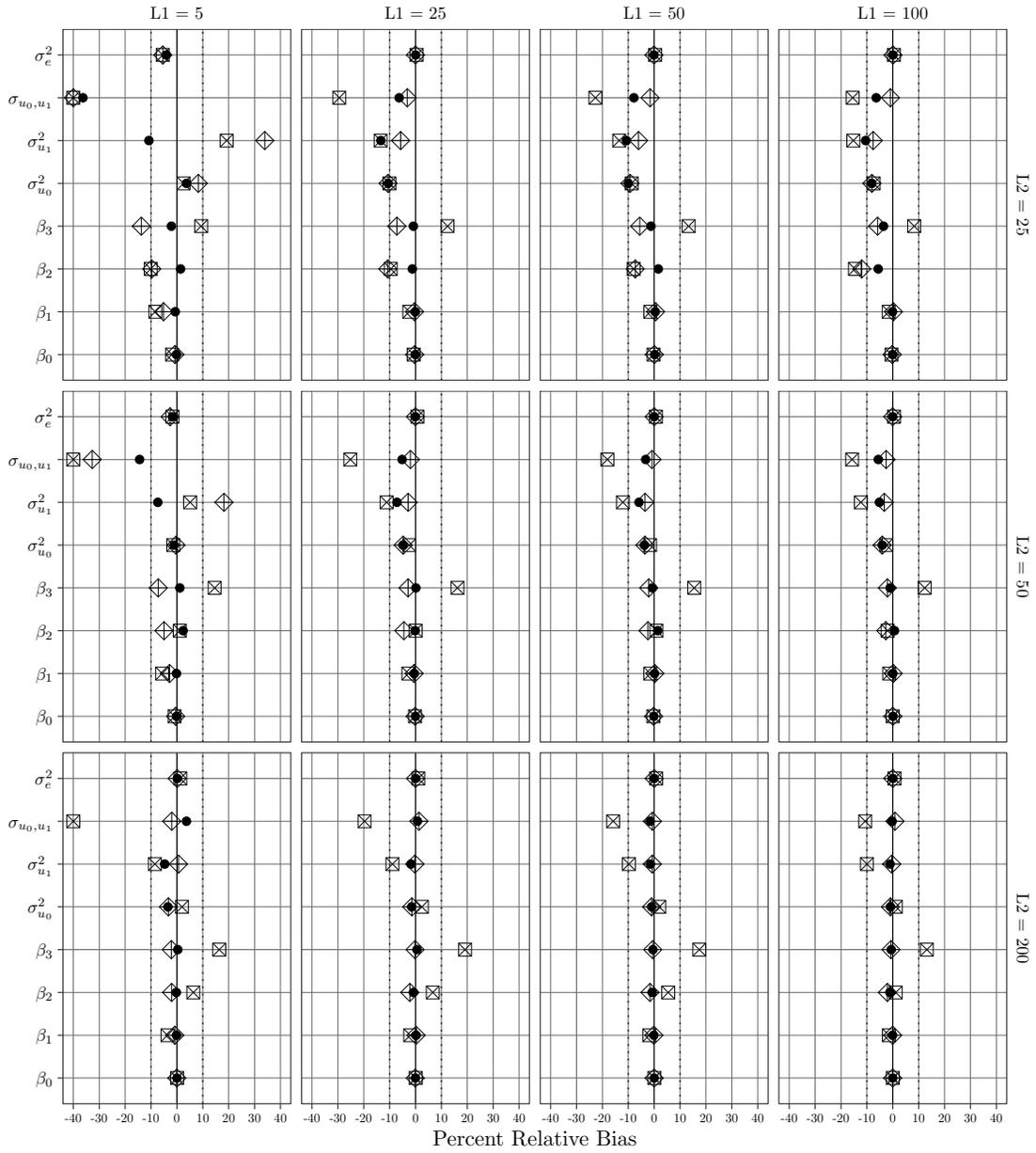
Figure 25: Scatterplot of twelve clusters for  $Y$  regressed on  $X$  for  $\chi^2(15)$  level-1 residual.

## APPENDIX B

### Simulation 1: 15% Missing Data Rate Results

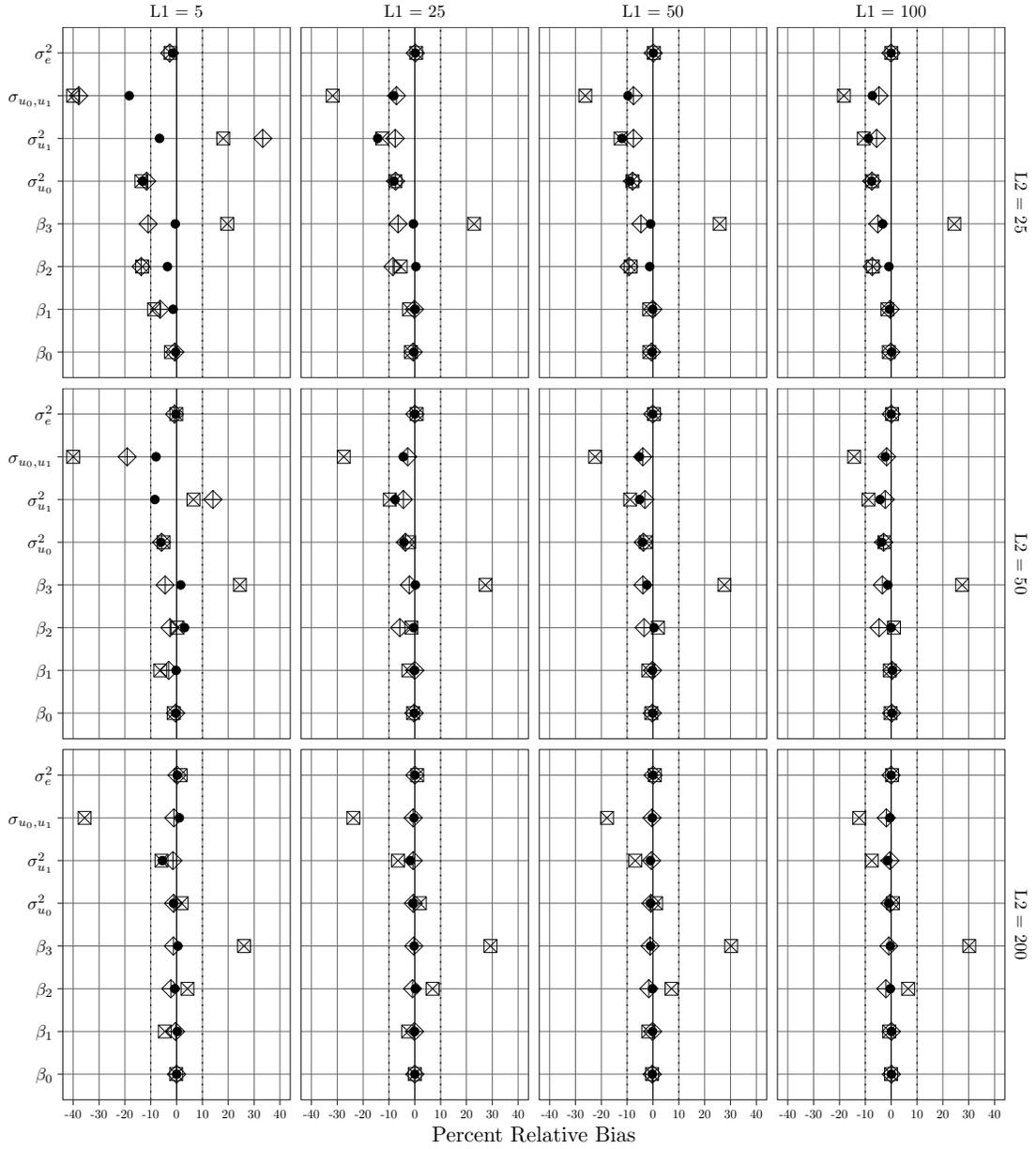
ICC = 0.10; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI



ICC = 0.50; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI

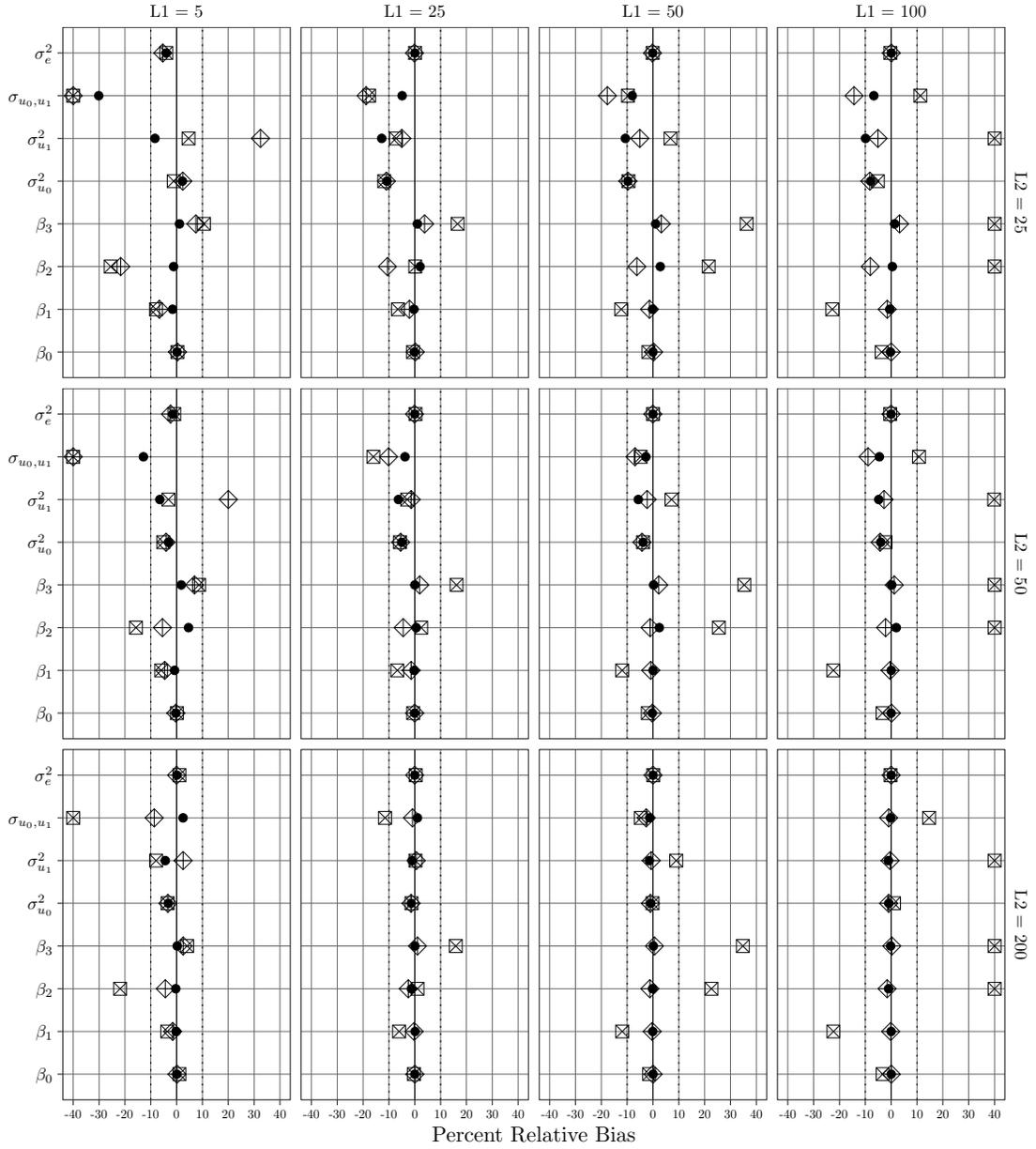


## APPENDIX C

### Simulation 2: 15% Missing Data Rate Results

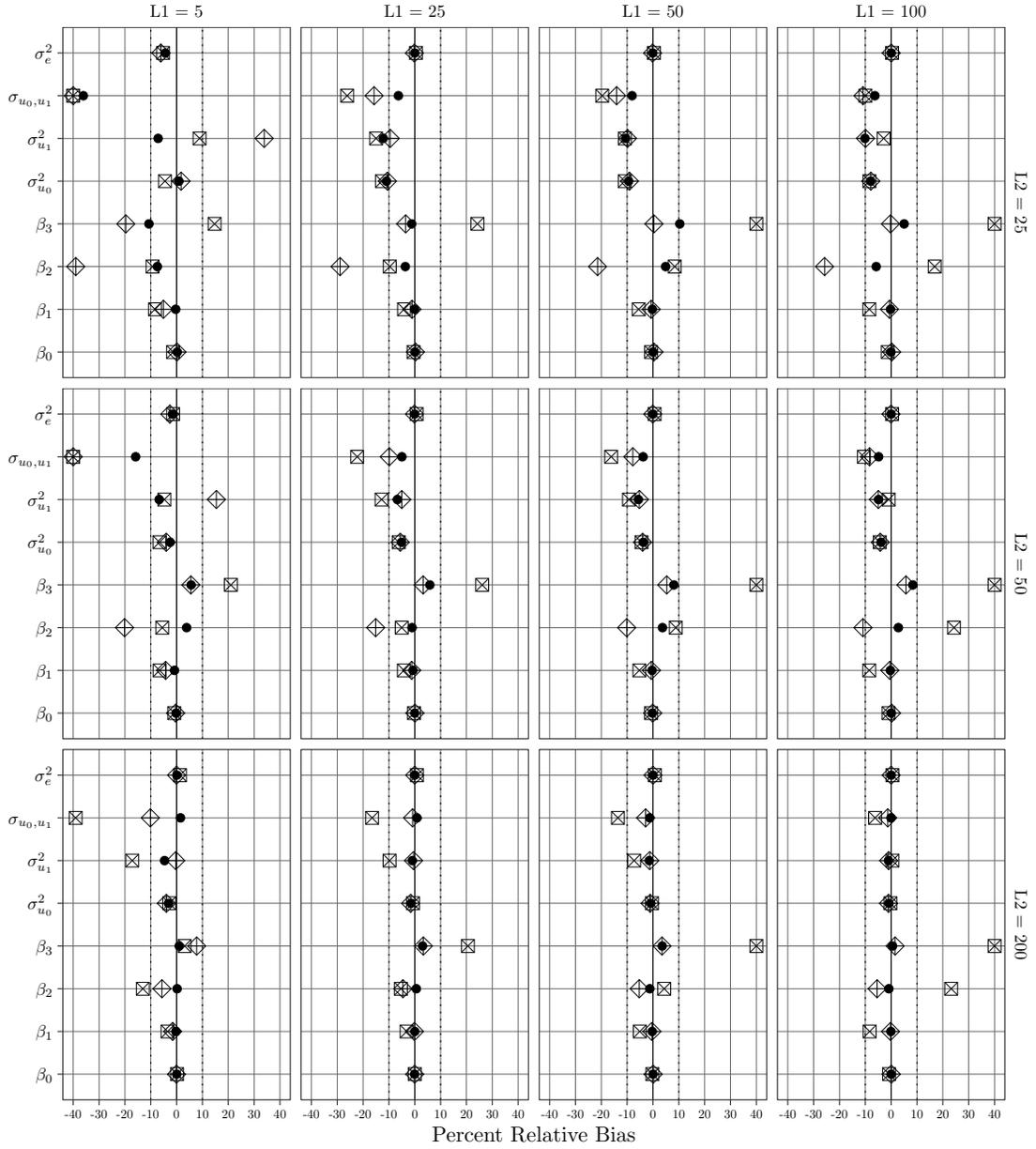
ICC = 0.10; Missing = 15%; Split = 0.5

● CMP    ⊠ JAV    ◇ MBI



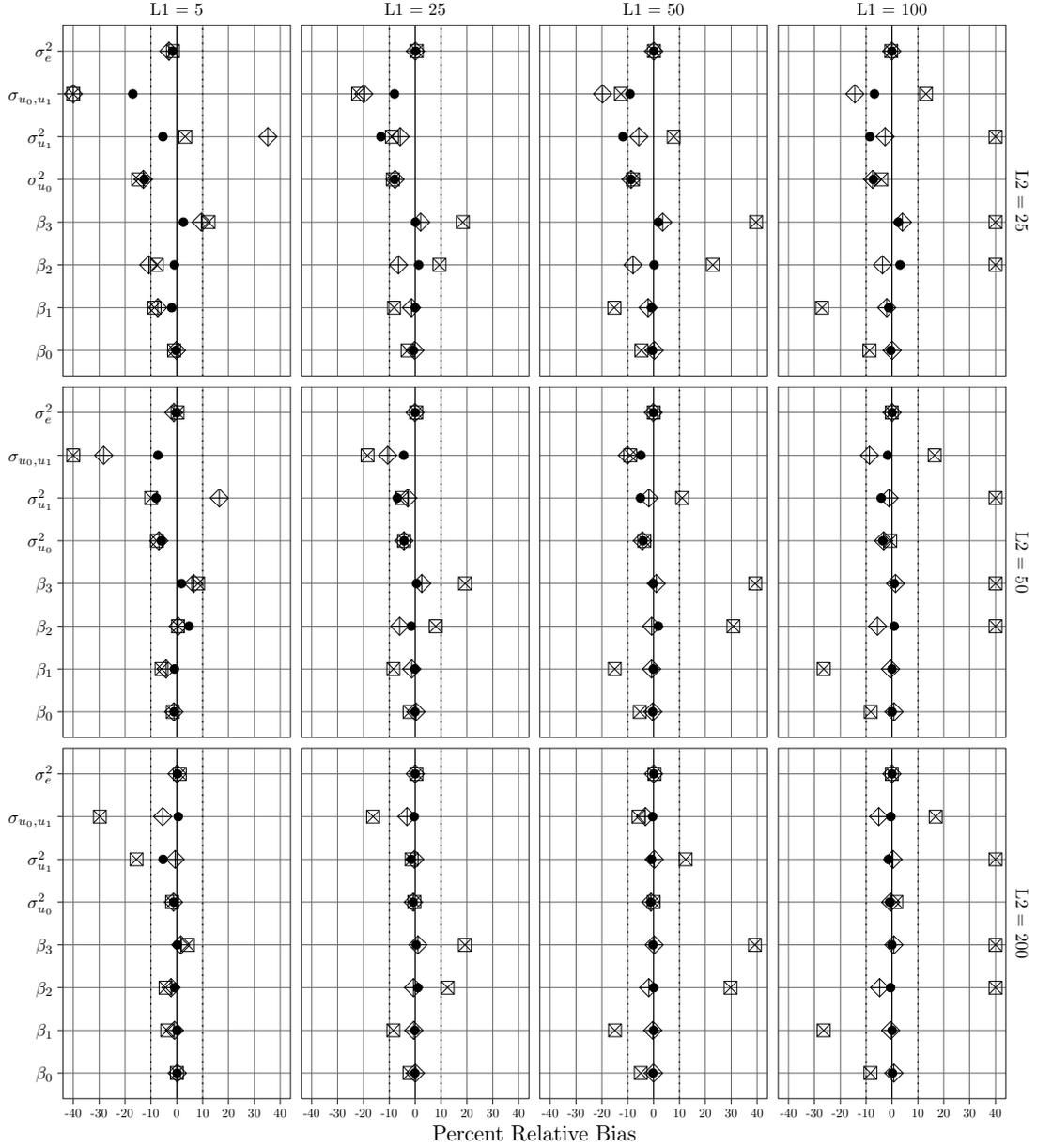
ICC = 0.10; Missing = 15%; Split = 0.8

● CMP    ⊠ JAV    ◇ MBI



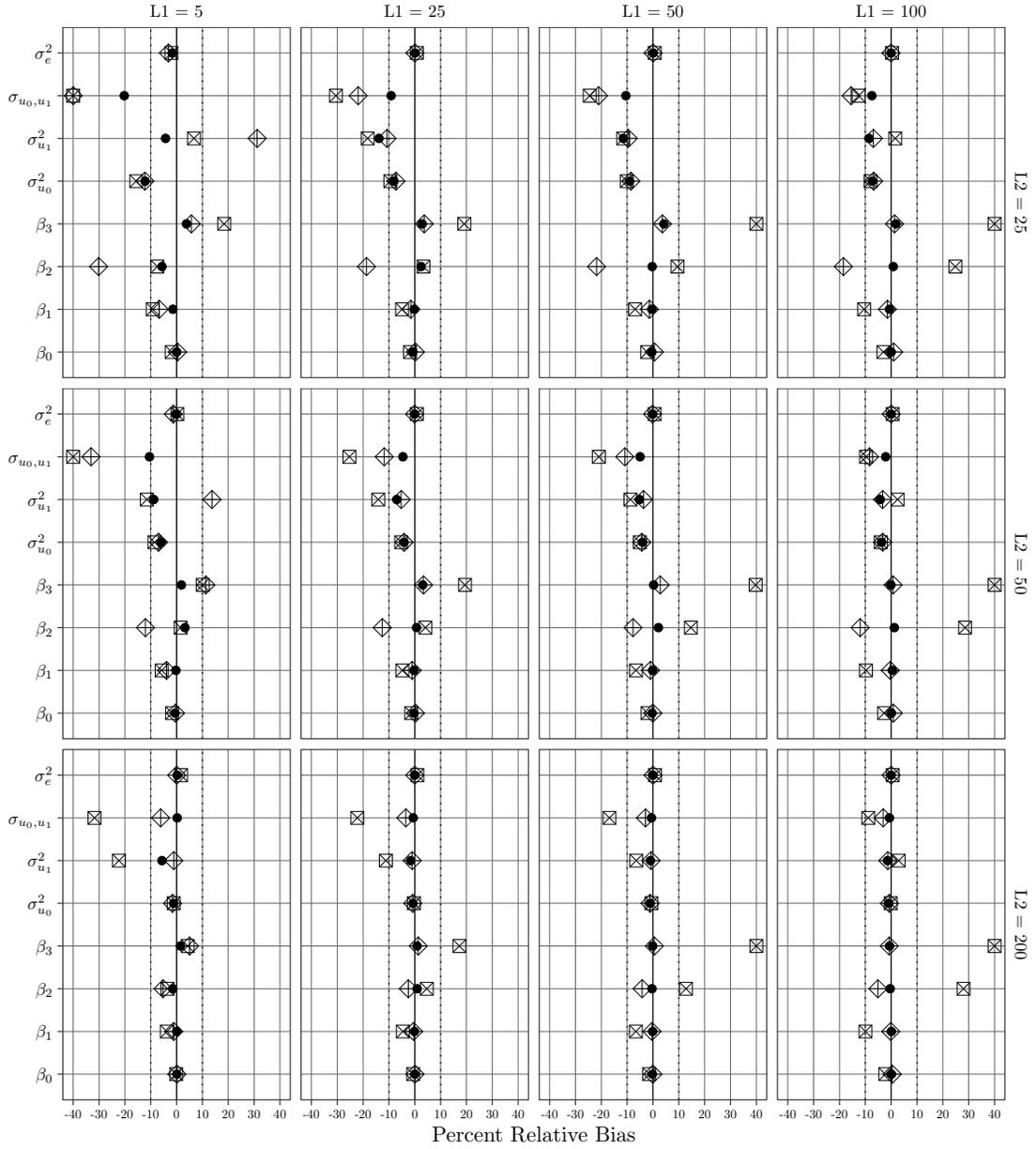
ICC = 0.50; Missing = 15%; Split = 0.5

● CMP    ⊠ JAV    ◇ MBI



ICC = 0.50; Missing = 15%; Split = 0.8

● CMP    ⊠ JAV    ◇ MBI

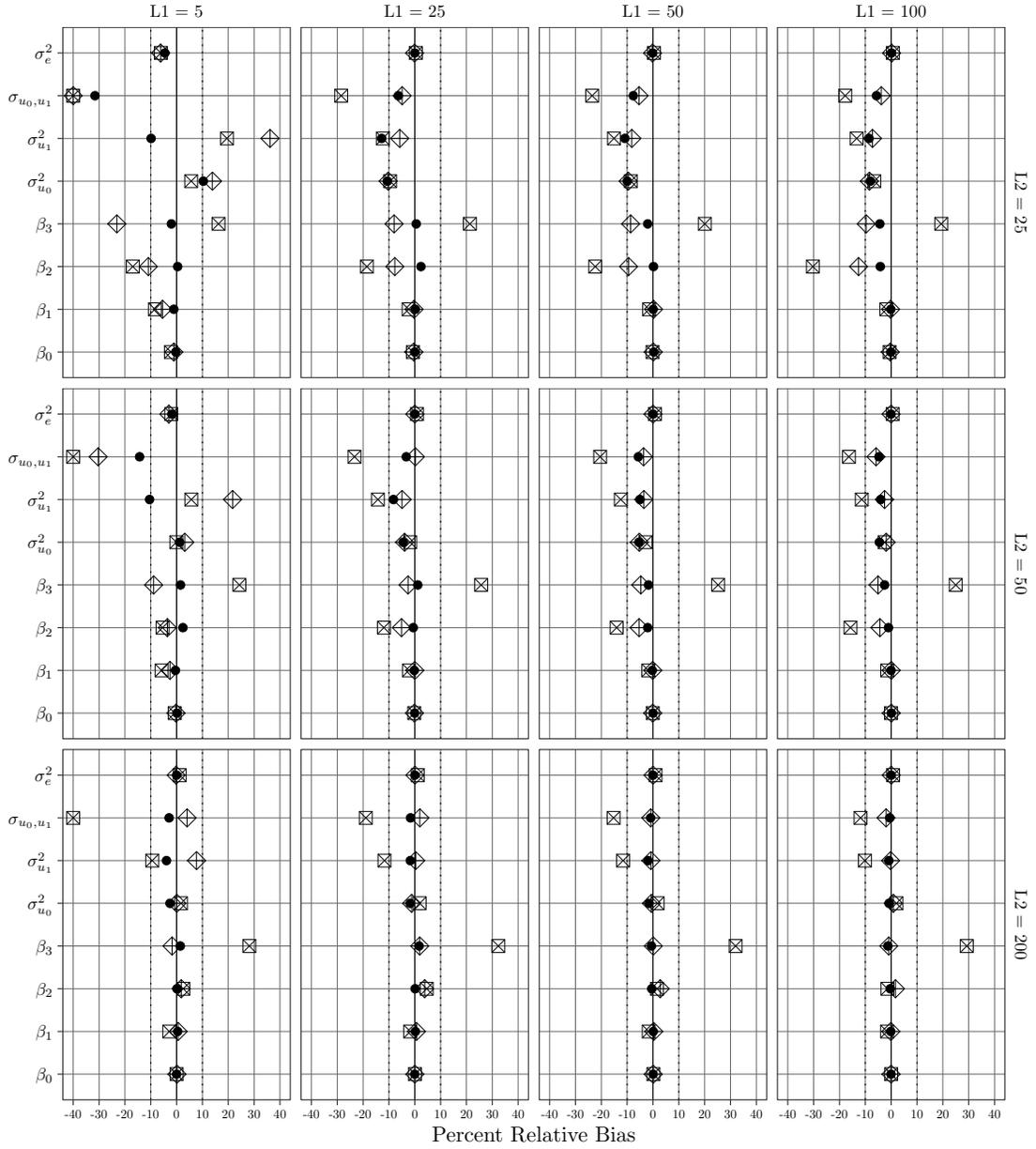


## APPENDIX D

### Simulation 3: 15% Missing Data Rate Results

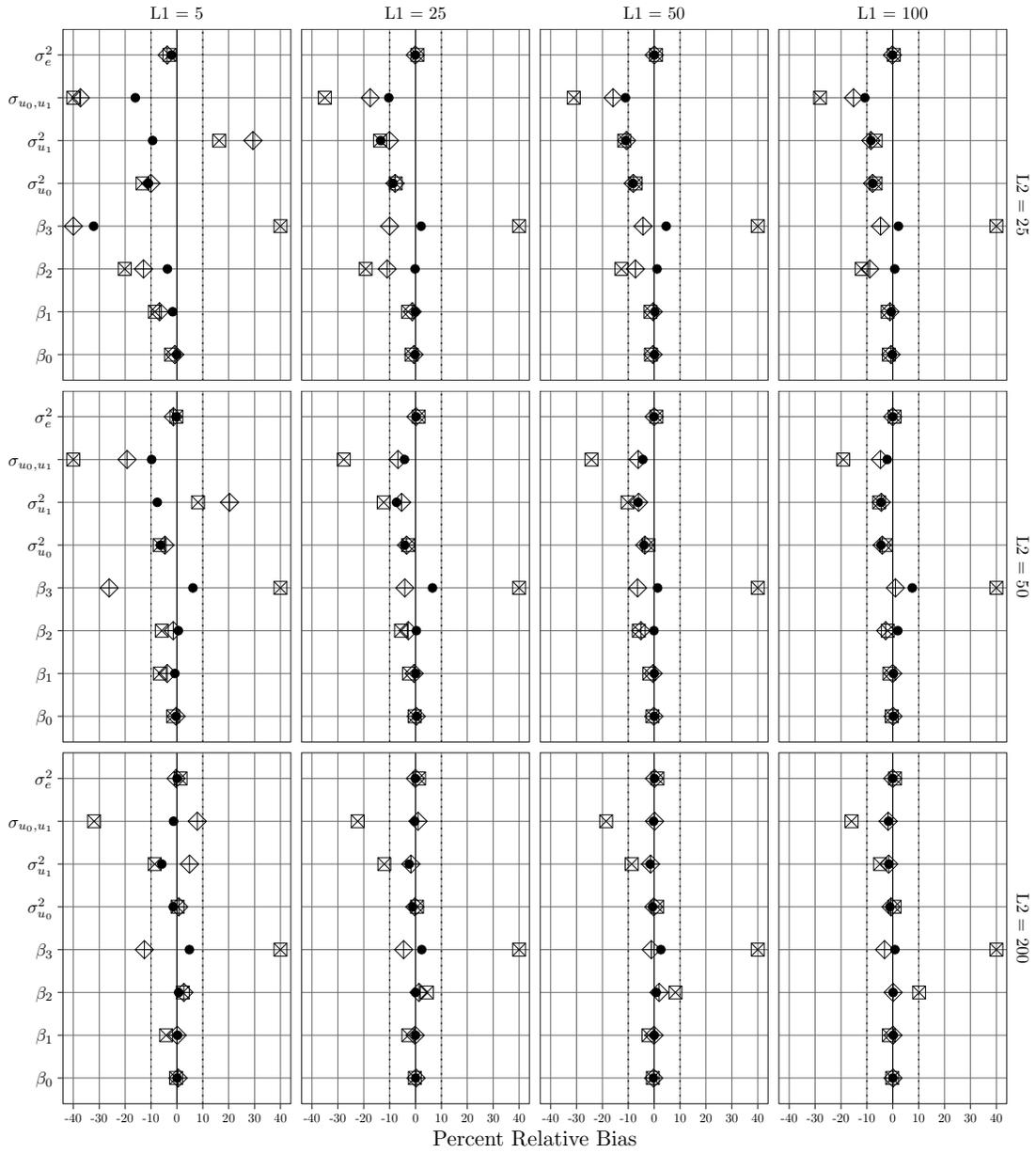
ICC = 0.10; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI



ICC = 0.50; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI

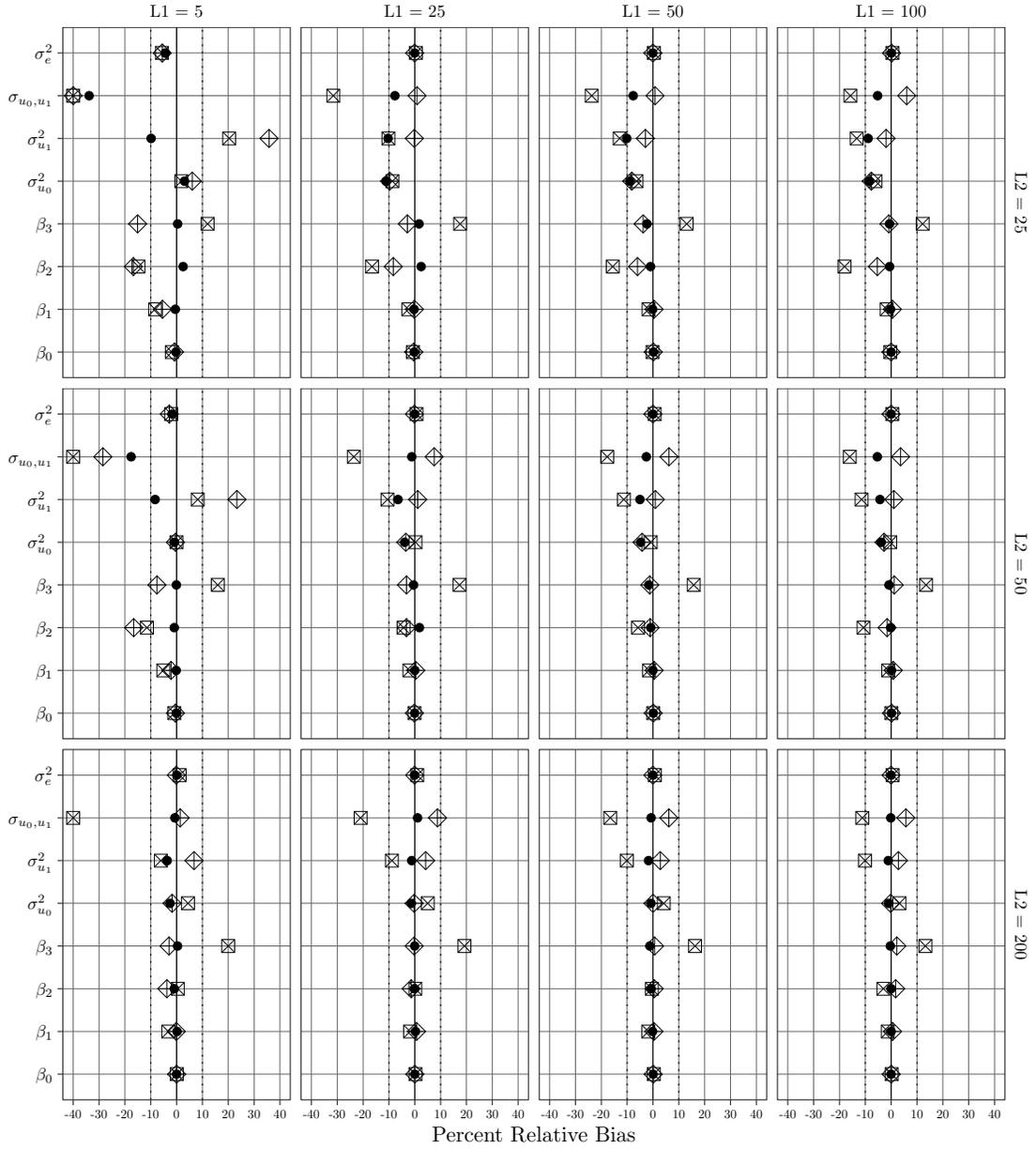


## APPENDIX E

### Simulation 4: 15% Missing Data Rate Results

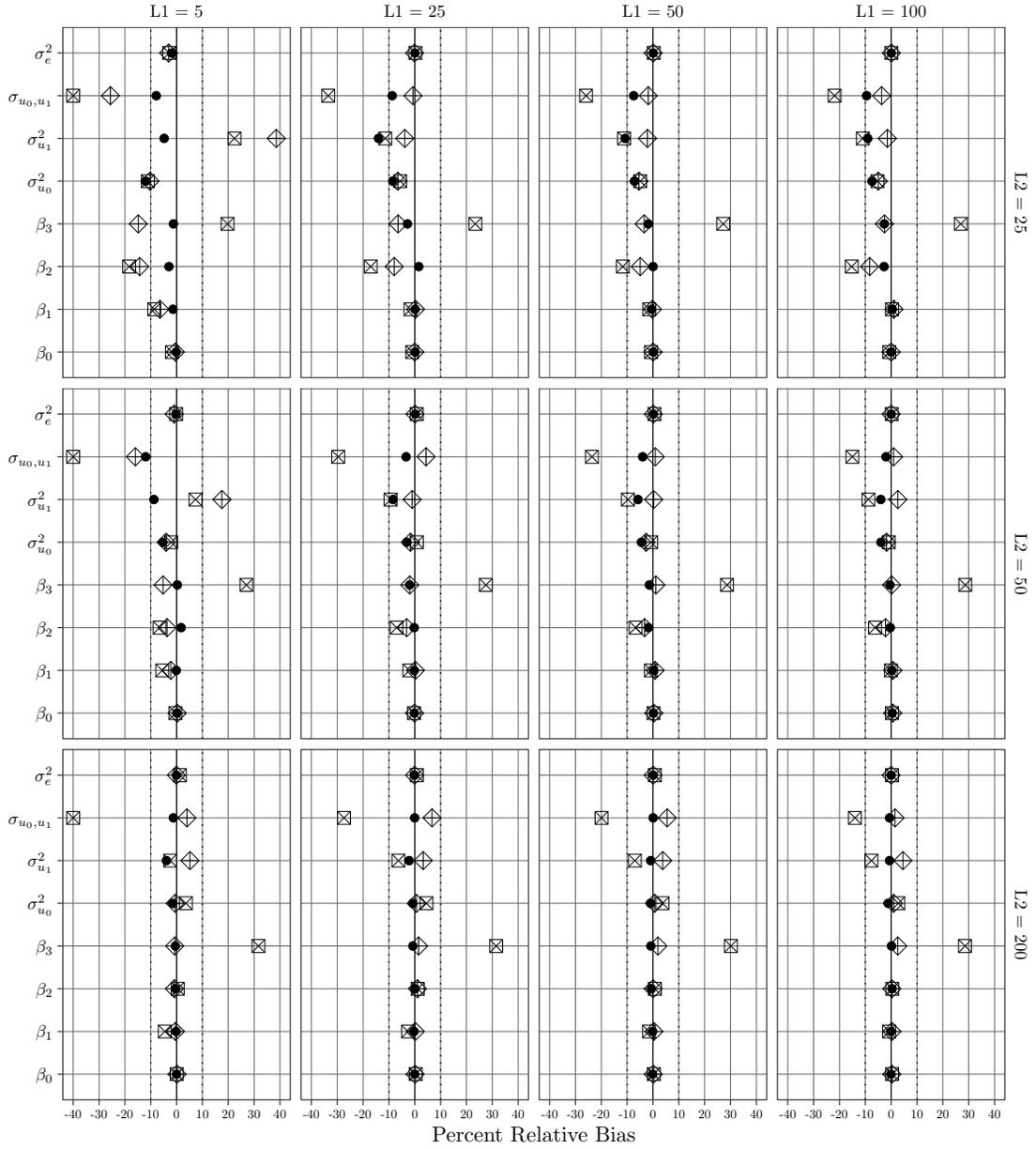
ICC = 0.10; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI



ICC = 0.50; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI

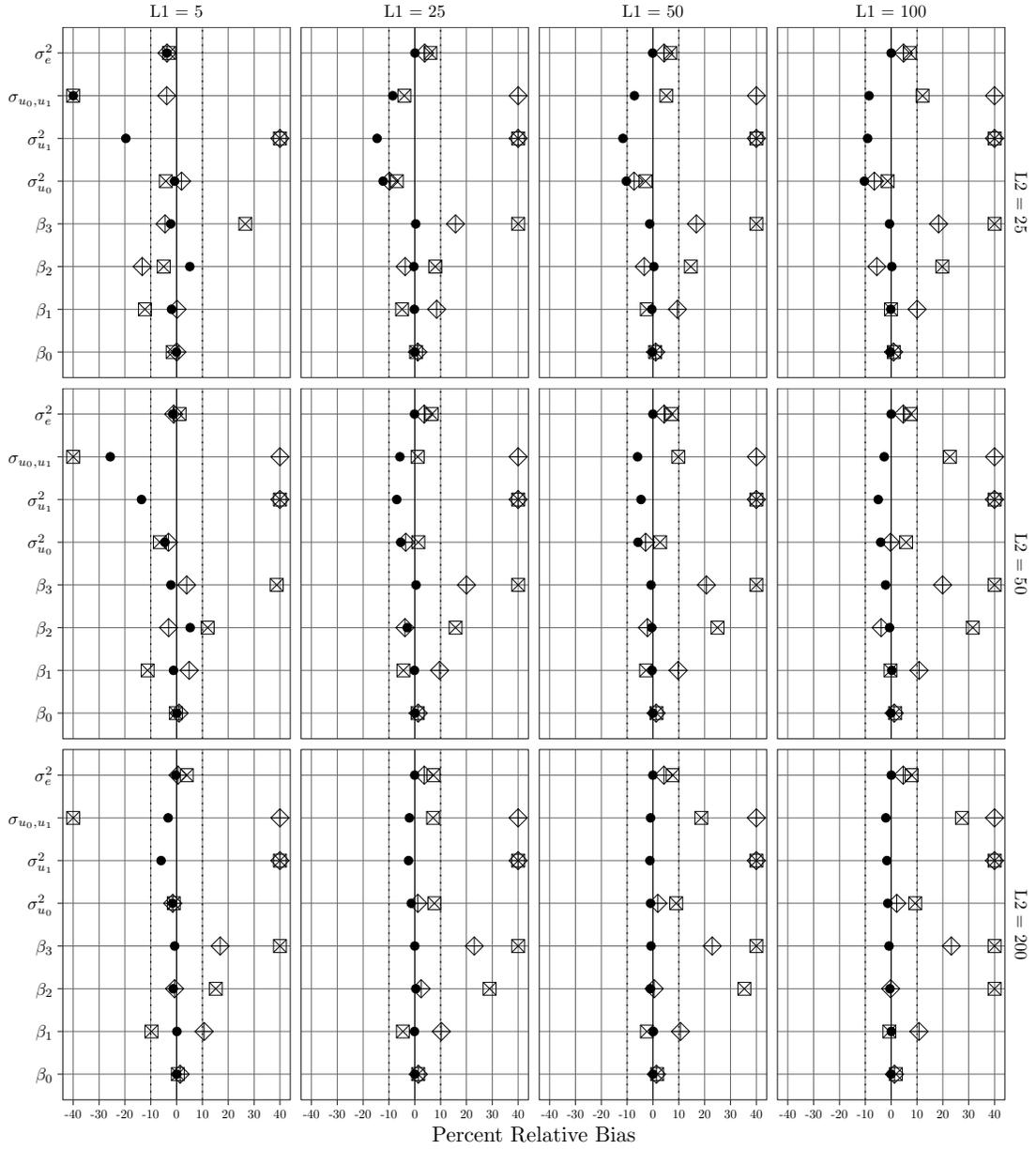


## APPENDIX F

### Simulation 5: 15% Missing Data Rate Results

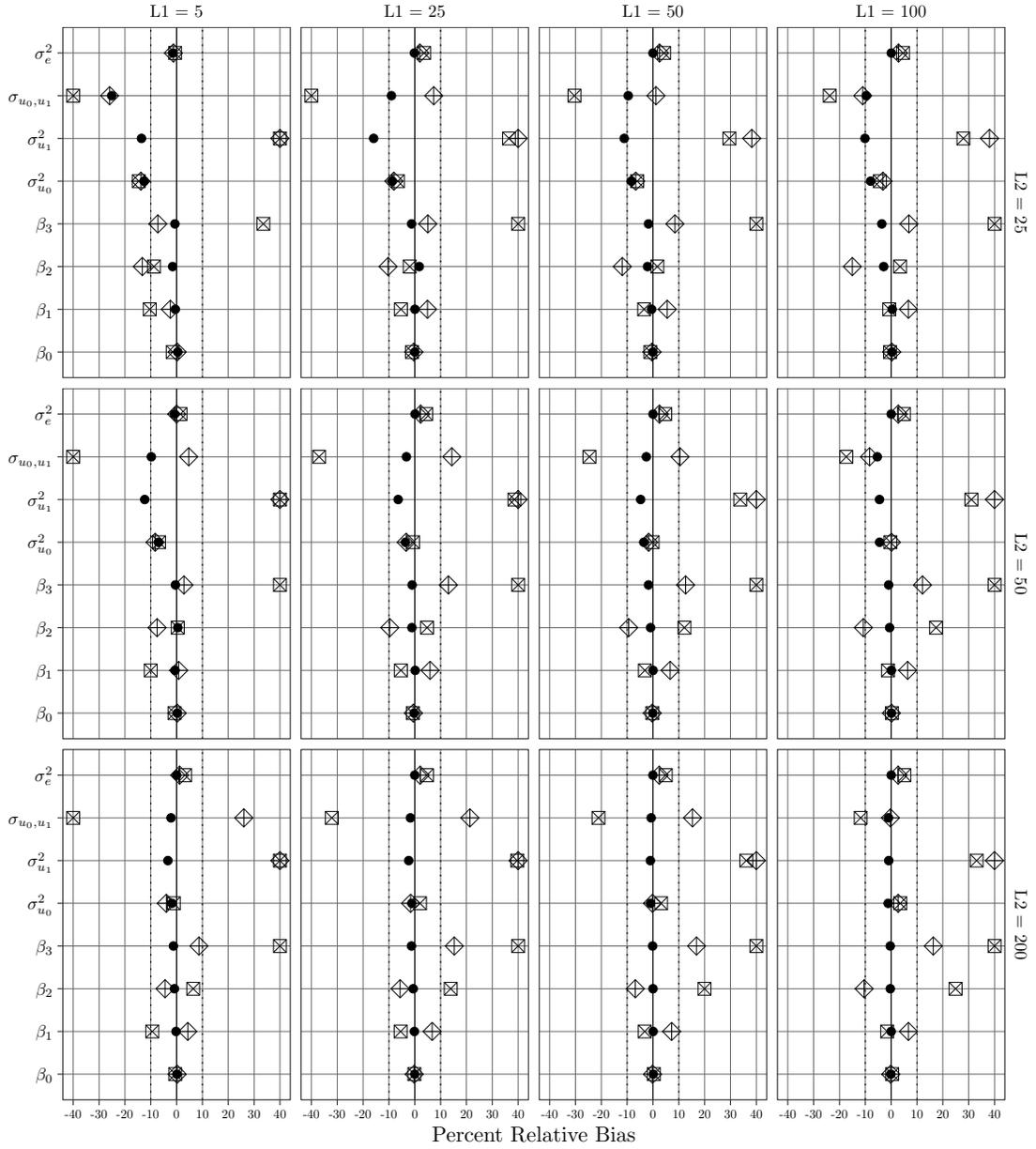
ICC = 0.10; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI



ICC = 0.50; Missing = 15%

● CMP    ⊠ JAV    ◇ MBI

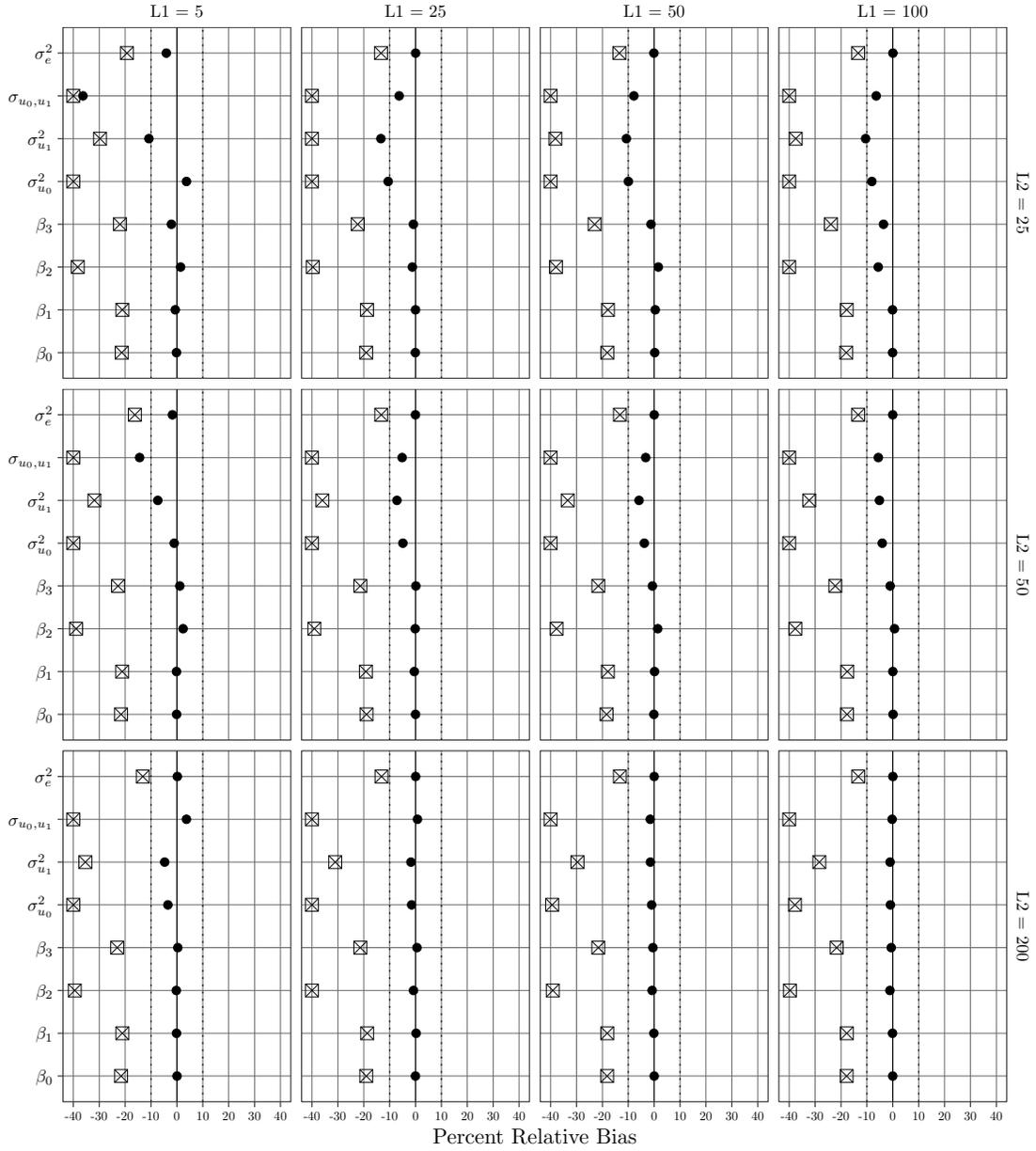


## APPENDIX G

### Simulation 1: Listwise Deletion Results

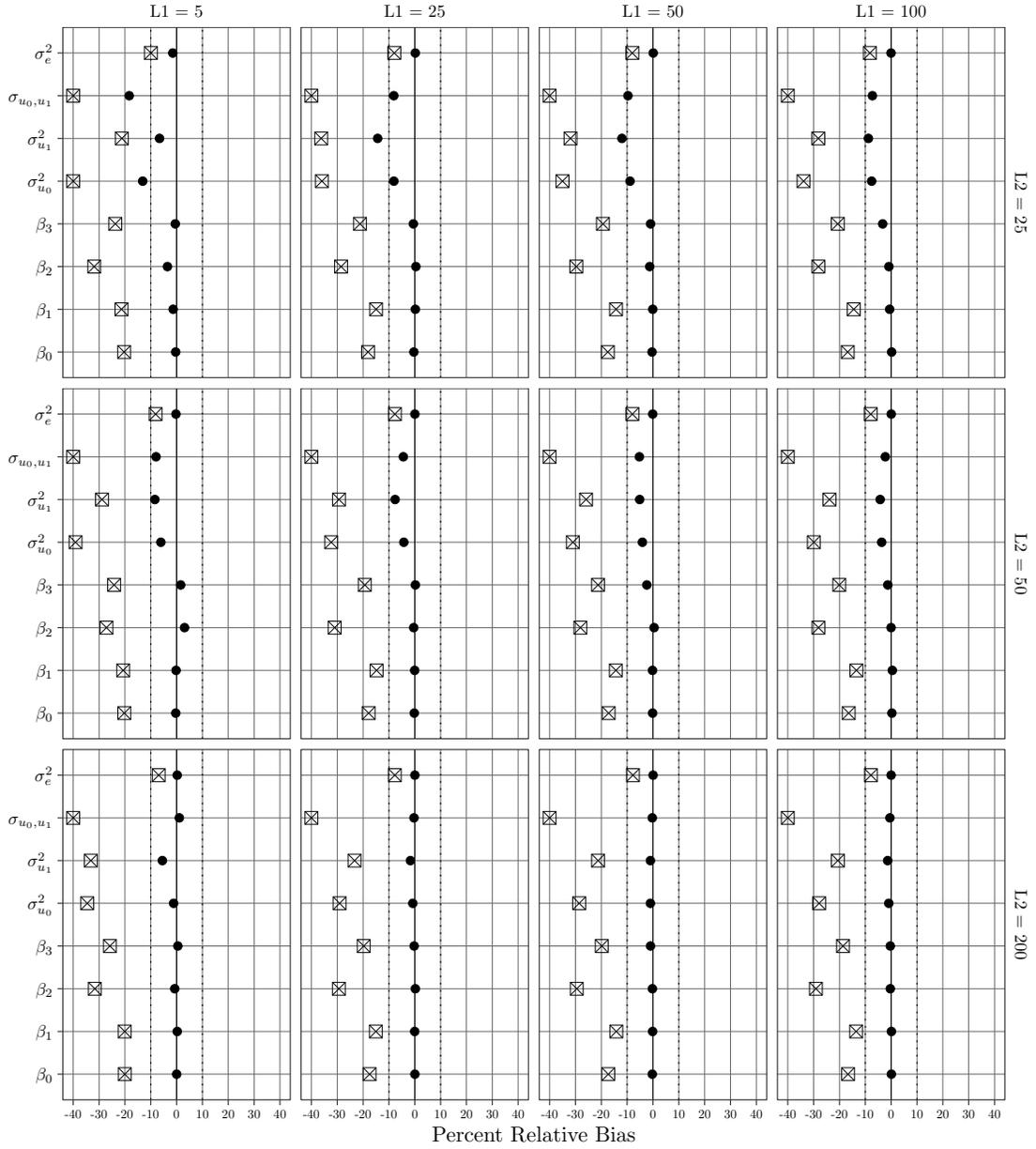
ICC = 0.10; Missing = 15%

● CMP ☒ LIST



ICC = 0.50; Missing = 15%

● CMP    ⊠ LIST

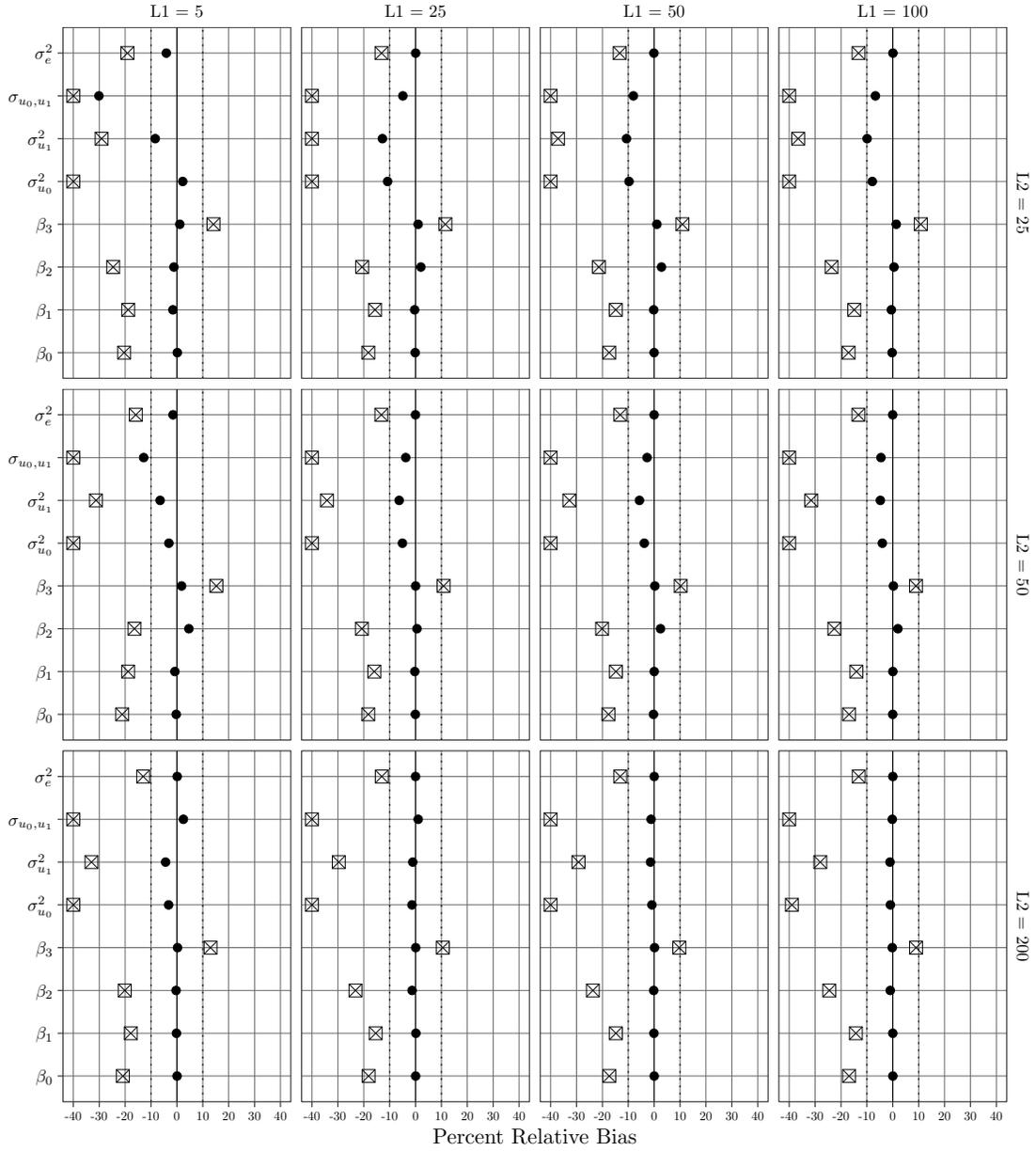


## APPENDIX H

### Simulation 2: Listwise Deletion Results

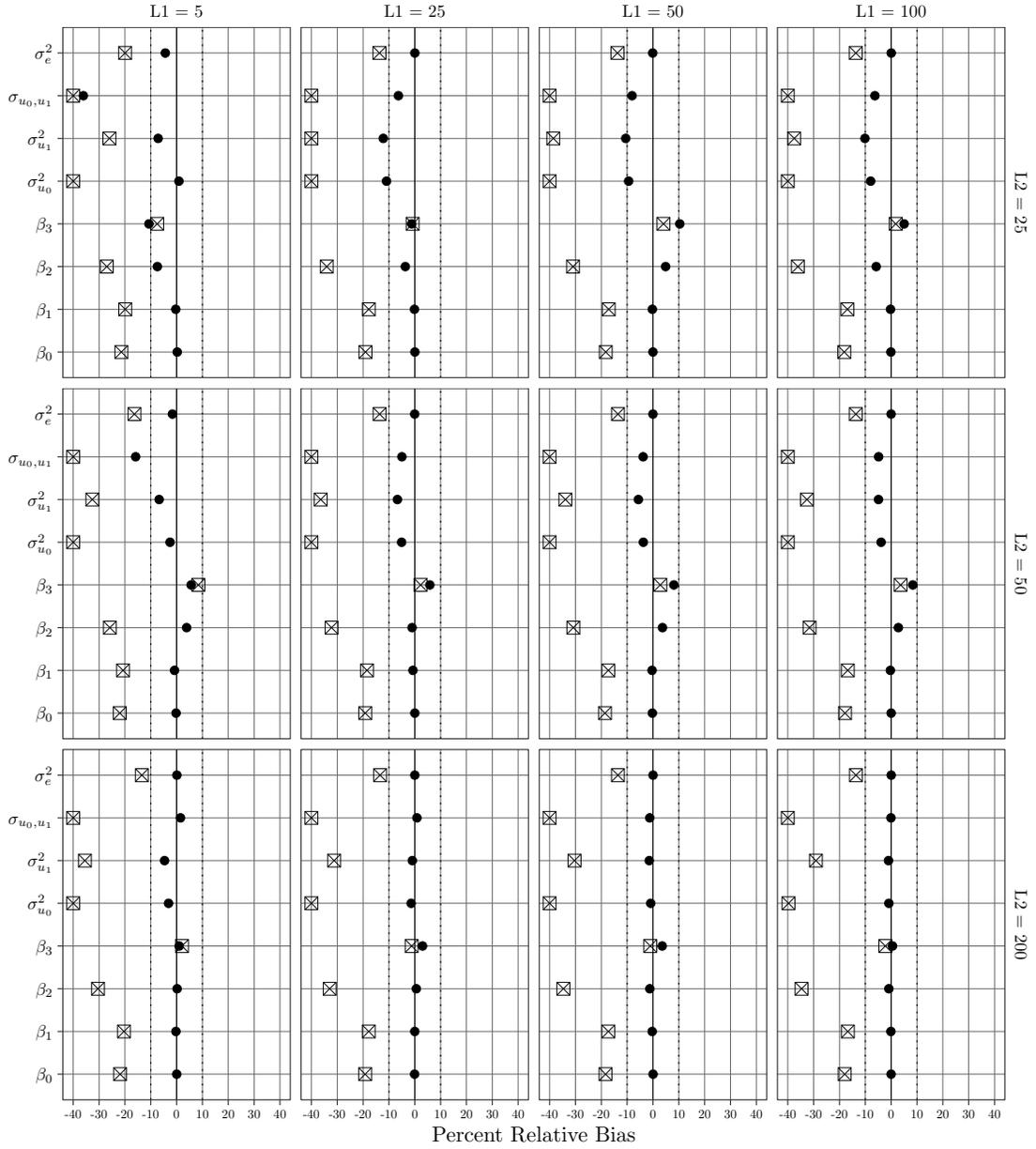
ICC = 0.10; Missing = 15%; Split = 0.5

● CMP    ☒ LIST



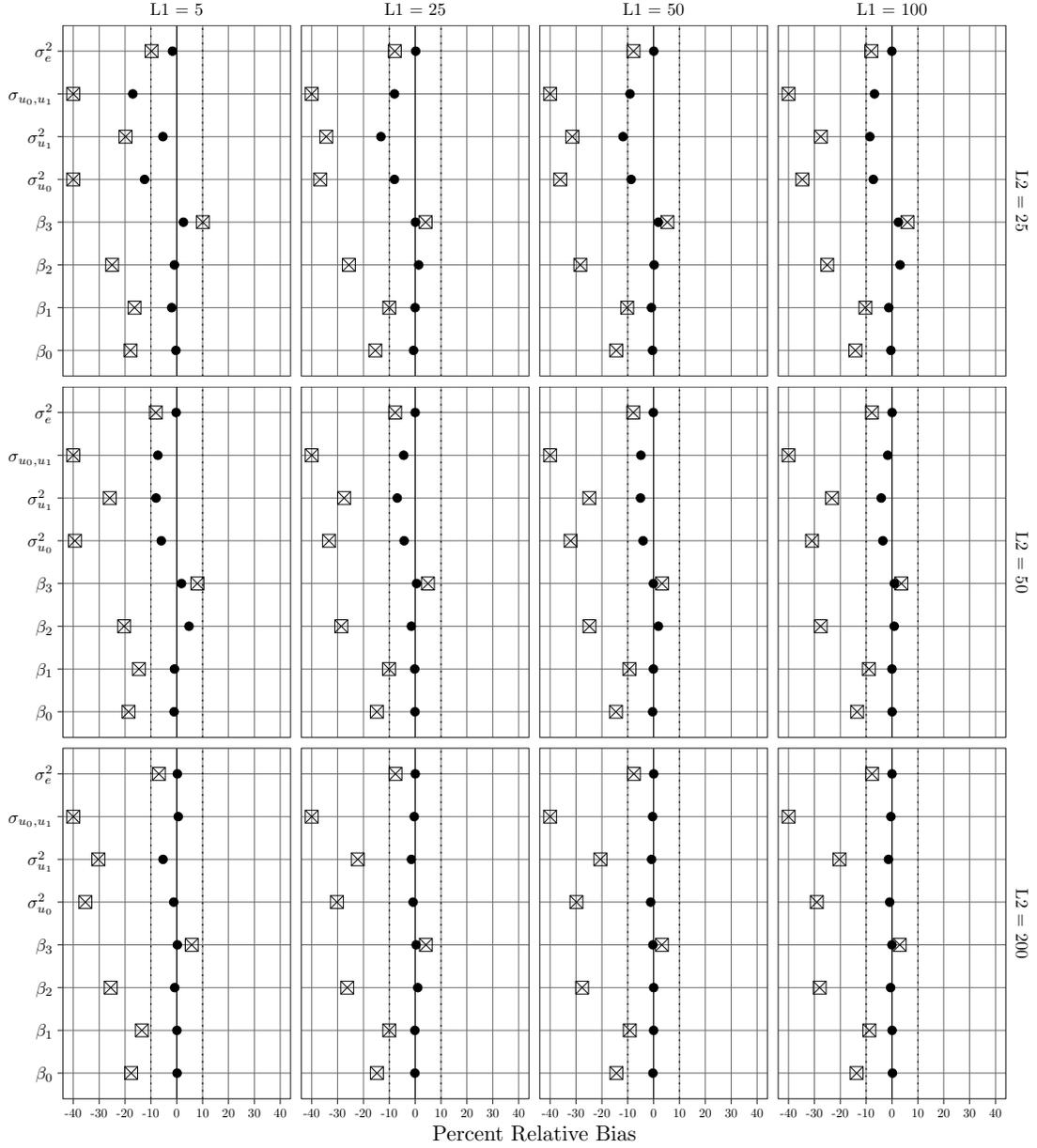
ICC = 0.10; Missing = 15%; Split = 0.8

● CMP    ⊠ LIST



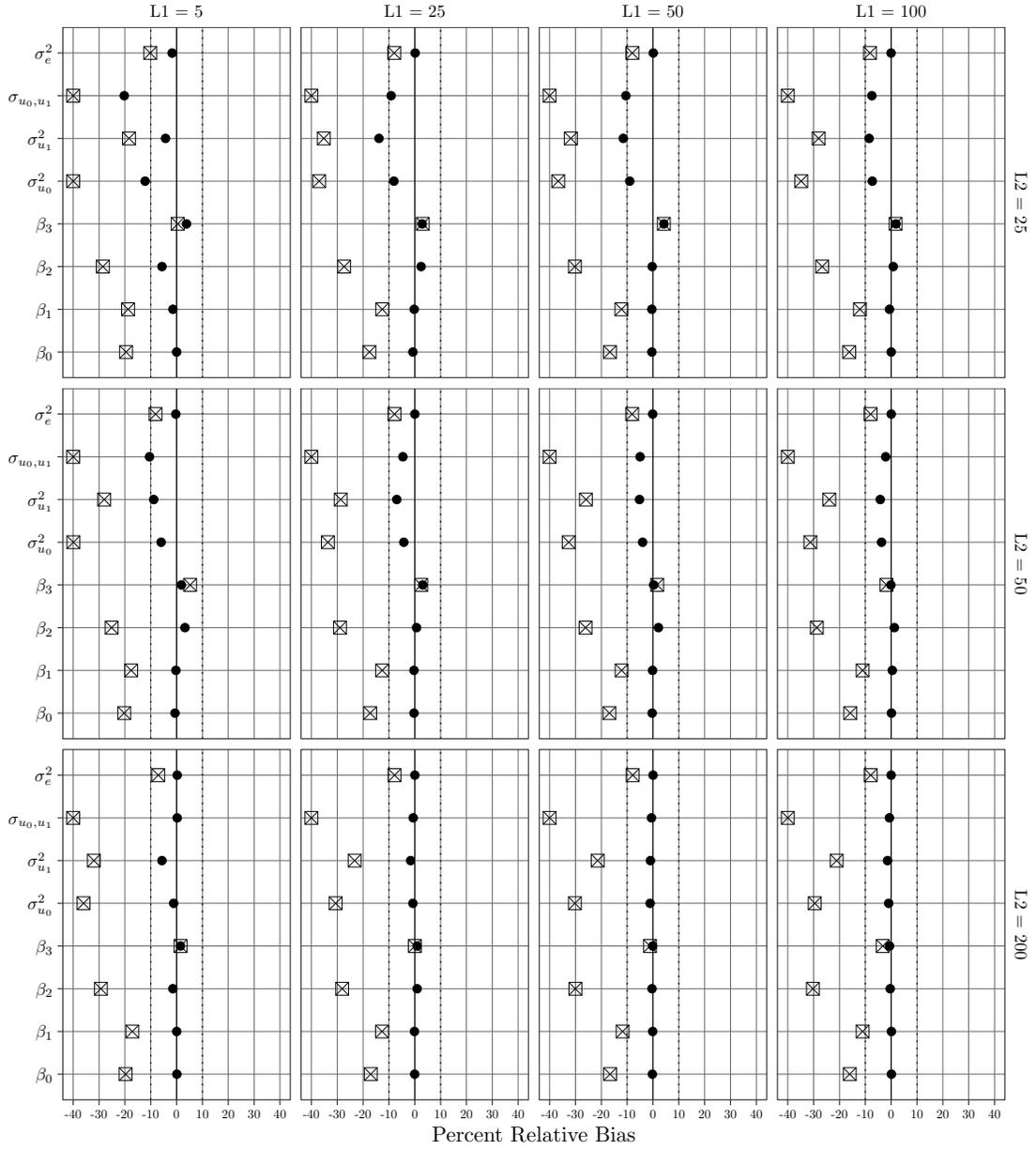
ICC = 0.50; Missing = 15%; Split = 0.5

● CMP ☒ LIST



ICC = 0.50; Missing = 15%; Split = 0.8

● CMP    ☒ LIST

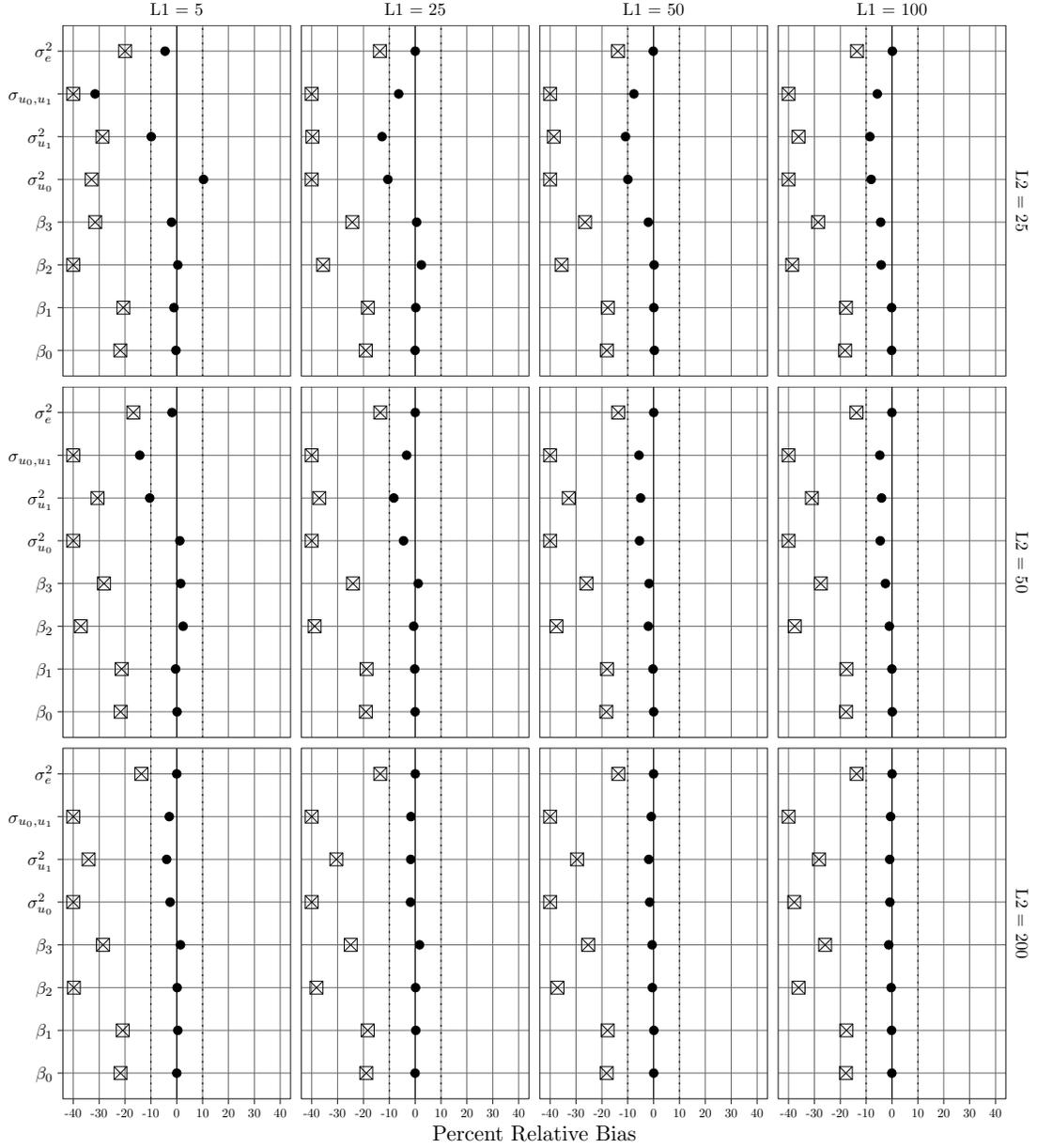


## APPENDIX I

### Simulation 3: Listwise Deletion Results

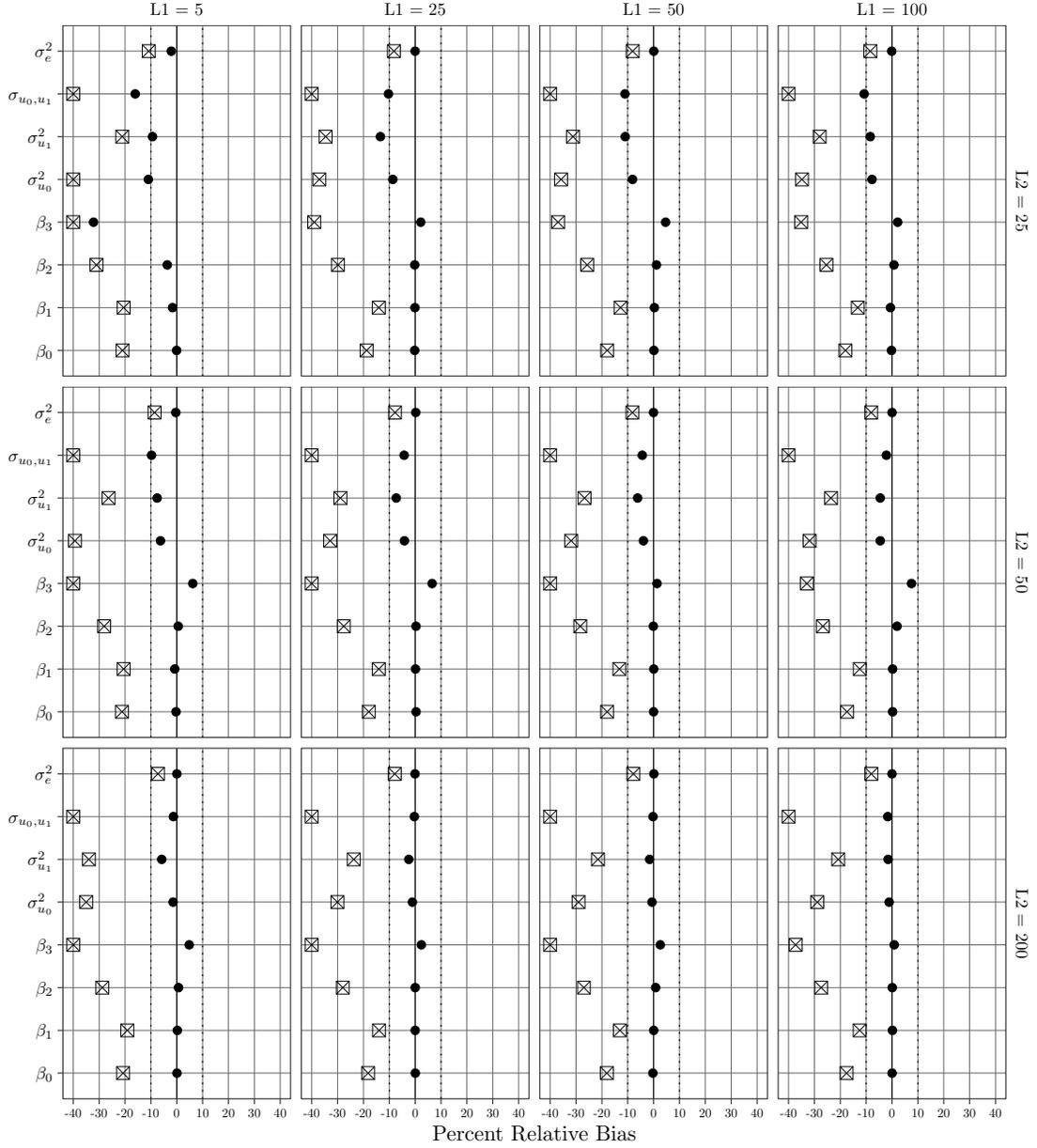
ICC = 0.10; Missing = 15%

● CMP    ☒ LIST



ICC = 0.50; Missing = 15%

● CMP    ⊠ LIST

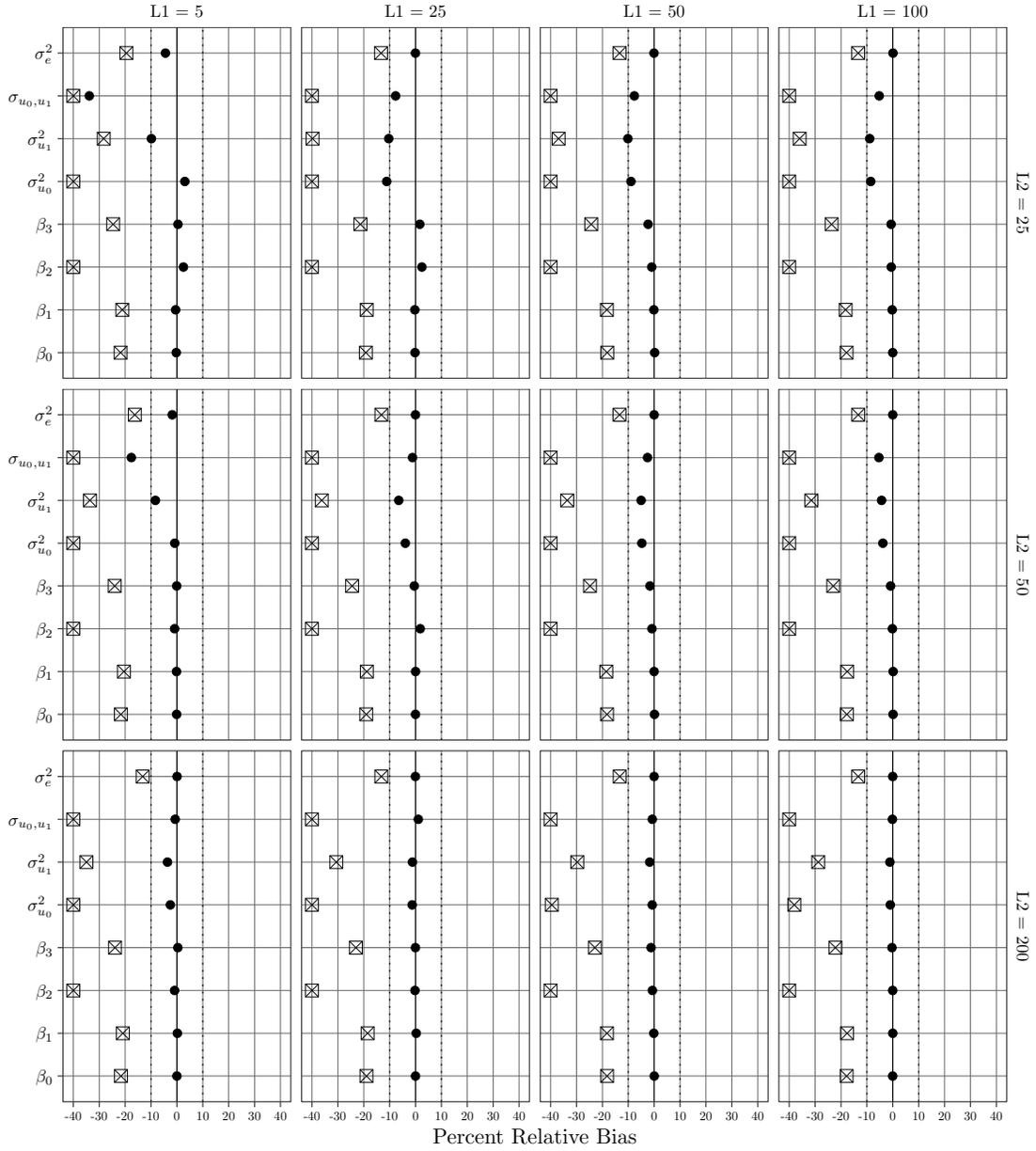


## APPENDIX J

### Simulation 4: Listwise Deletion Results

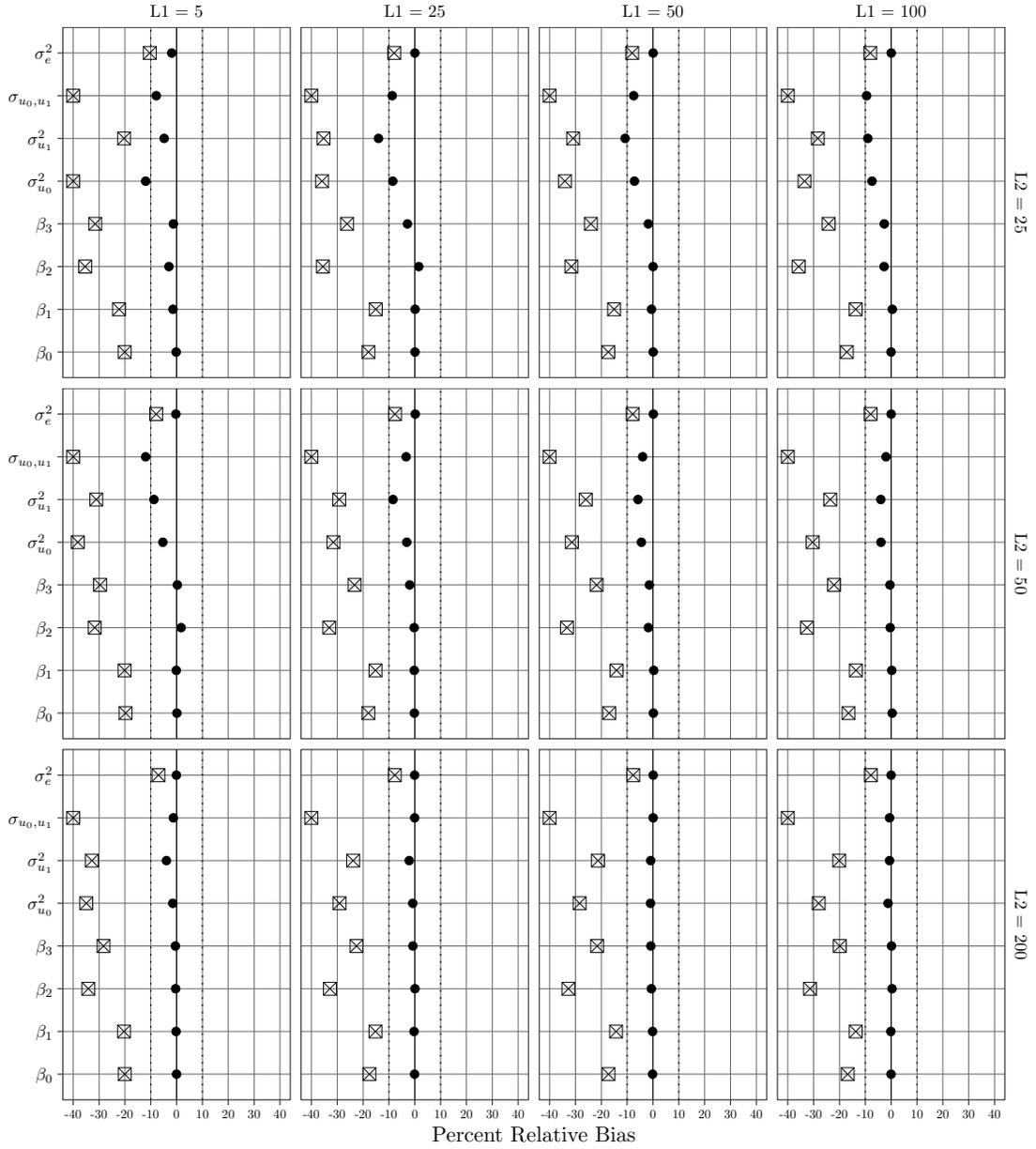
ICC = 0.10; Missing = 15%

● CMP    ⊠ LIST



ICC = 0.50; Missing = 15%

● CMP    ⊠ LIST

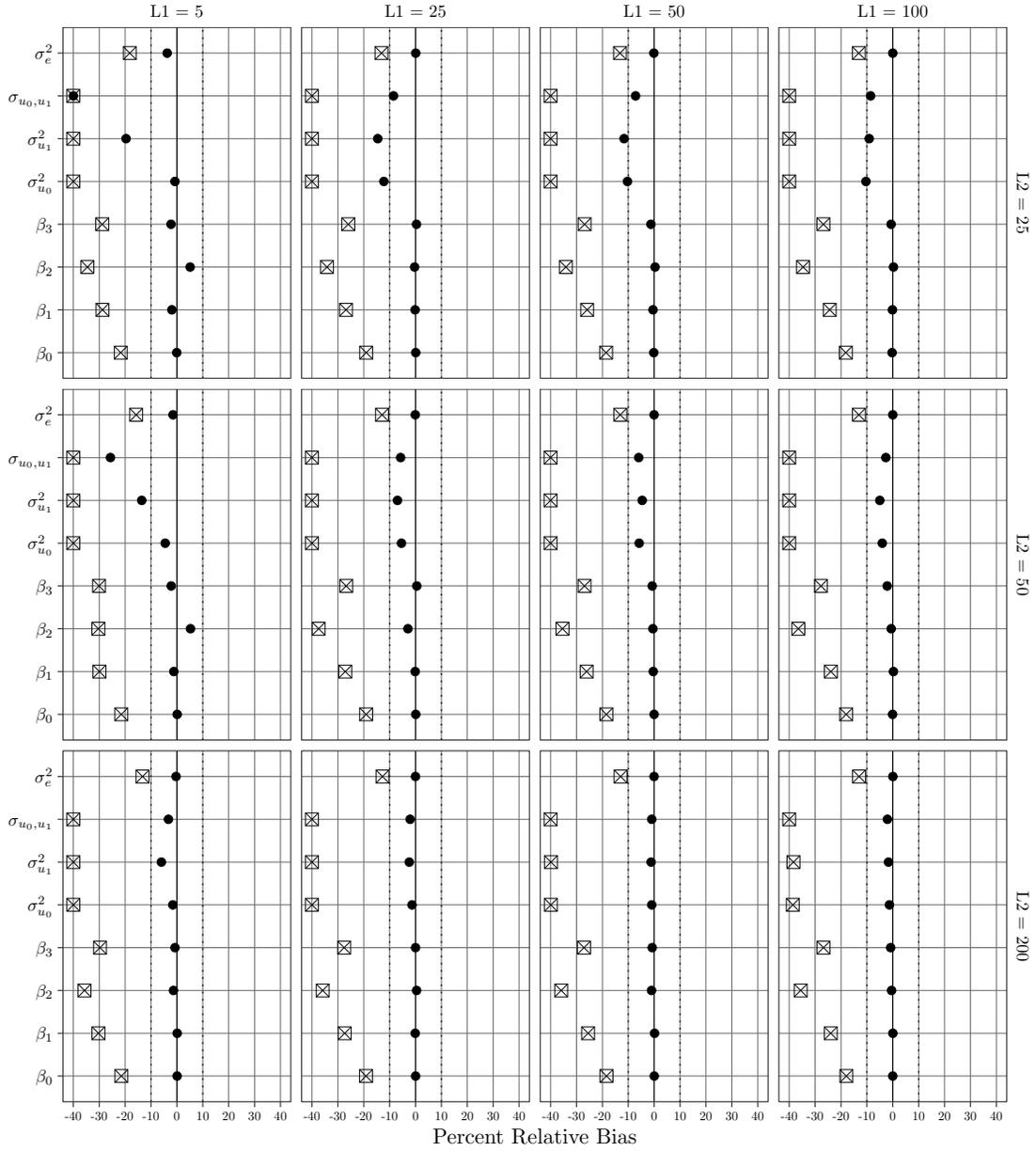


## APPENDIX K

### Simulation 5: Listwise Deletion Results

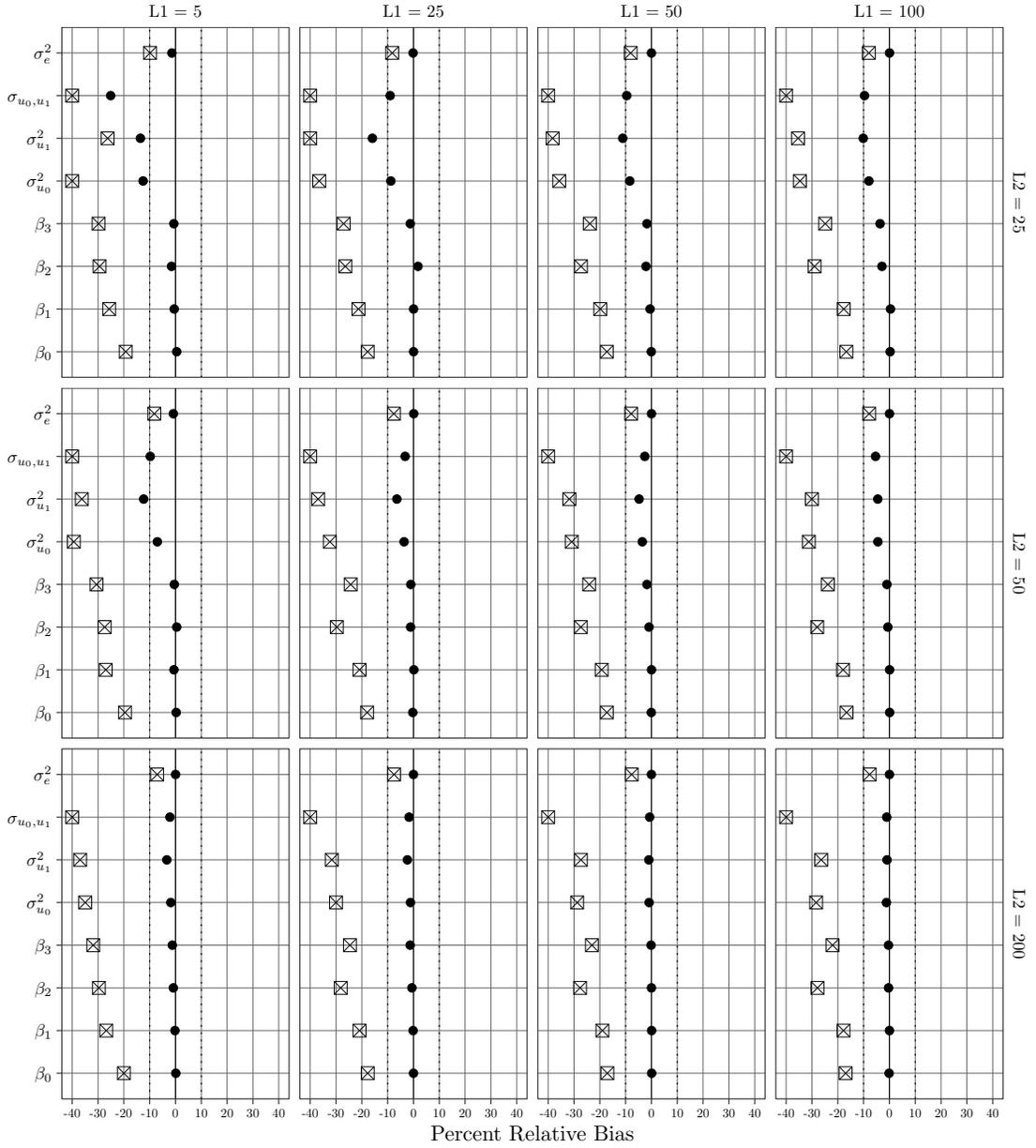
ICC = 0.10; Missing = 15%

● CMP    ⊠ LIST



ICC = 0.50; Missing = 15%

● CMP    ☒ LIST



## REFERENCES

- Agresti, A. (2012). *Analysis of ordinal categorical data* (3rd ed.). Hoboken, NJ: Wiley.
- Albert, J. H., & Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, *88*, 669679.
- Allison, P. D. (2002). *Missing data*. Newbury Park, CA: Sage.
- Bartlett, J. W., Seaman, S. R., White, I. R., & Carpenter, J. R. (2012). Accommodating the model of interest within the fully conditional specification multiple imputation framework.
- Bartlett, J. W., Seaman, S. R., White, I. R., & Carpenter, J. R. (2014). Multiple imputation of covariates by fully conditional specification: Accommodating the substantive model. *Statistical Methods and Research*, *0*, 1-26. doi: 10.1177/0962280214521348
- Bradley, J. V. (1978). Robustness? *British Journal of Mathematical and Statistical Psychology*, *31*, 144-152. doi: 10.1111/j.2044-8317.1978.tb00581.x
- Browne, W. J. (1998). *Applying mcmc methods to multi-level models* (Unpublished doctoral dissertation).
- Carpenter, J., & Kenward, M. G. (2013). *Multiple imputation and its application*. West Sussex, UK: Wiley.
- Chung, Y., Gelman, A., Rabe-Hesketh, S., Liu, J., & Dorie, V. (2015). Weakly informative prior for point estimation of covariance matrices in hierarchical models. *Journal of Educational and Behavioral Statistics*, *40*(2), 136–157.
- Enders, C. K., Baraldi, A. N., & Cham, H. (2014). Estimating interaction effects with incomplete predictor variables. *Psychological methods*, *19*(1), 39.
- Enders, C. K., Du, H., & Keller, B. T. (in press). A model-based imputation procedure for multilevel regression models with random coefficients, interaction effects, and non-linear terms. *Psychological methods*.
- Enders, C. K., & Gottschall, A. C. (2011). Multiple imputation strategies for multiple group structural equation models. *Structural Equation Modeling*, *18*(1), 35–54.
- Enders, C. K., Hayes, T., & Du, H. (2018). A comparison of multilevel imputation schemes for

- random coefficient models: Fully conditional specification and joint model imputation with random covariance matrices. *Multivariate behavioral research*, 53(5), 695–713.
- Enders, C. K., Keller, B. T., & Levy, R. (2017). A fully conditional specification approach to multilevel imputation of categorical and continuous variables. *Psychological Methods*, Advance online publication. doi: dx.doi.org/10.1037/met0000148
- Erler, N. S., Rizopoulos, D., Jaddoe, V. W., Franco, O. H., & Lesaffre, E. M. (2019). Bayesian imputation of time-varying covariates in linear mixed models. *Statistical methods in medical research*, 28(2), 555–568.
- Erler, N. S., Rizopoulos, D., Rosmalen, J. v., Jaddoe, V. W., Franco, O. H., & Lesaffre, E. M. (2016). Dealing with missing covariates in epidemiologic studies: a comparison between multiple imputation and a full bayesian approach. *Statistics in medicine*, 35(17), 2955–2974.
- Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper). *Bayesian analysis*, 1(3), 515–534.
- Gelman, A., & Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press.
- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4), 457–472.
- Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.
- Gilks, W. R., Best, N. G., & Tan, K. (1995). Adaptive rejection metropolis sampling within gibbs sampling. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 44(4), 455–472.
- Goldstein, H., Carpenter, J. R., & Browne, W. J. (2014). Fitting multilevel multivariate models with missing data in responses and covariates that may include interactions and non-linear terms. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 177(2), 553–564.
- Grund, S., Lüdtke, O., & Robitzsch, A. (2016). Multiple imputation of missing covariate values in multilevel models with random slopes: A cautionary note. *Behavior Research*

- Methods*, 48(2), 640–649.
- Grund, S., Lüdtke, O., & Robitzsch, A. (2018). Multiple imputation of missing data at level 2: A comparison of fully conditional and joint modeling in multilevel designs. *Journal of Educational and Behavioral Statistics*, 43(3), 316–353.
- Gulliford, M. C., Ukoumunne, O. C., & Chinn, S. (1999). Components of variance and intraclass correlations for the design of community-based surveys and intervention studies: data from the health survey for england 1994. *American Journal of Epidemiology*, 149(9), 876–883.
- Hedges, L. V., & Hedberg, E. C. (2007). Intraclass correlation values for planning group-randomized trials in education. *Educational Evaluation and Policy Analysis*, 29(1), 60–87.
- Hox, J. J., Moerbeek, M., & Van de Schoot, R. (2017). *Multilevel analysis: Techniques and applications*. Routledge.
- Ibrahim, J. G., Chen, M.-H., & Lipsitz, S. R. (2002). Bayesian methods for generalized linear models with covariates missing at random. *Canadian Journal of Statistics*, 30(1), 55–78.
- Johnson, V. E., & Albert, J. H. (2006). *Ordinal data modeling*. Springer Science & Business Media.
- Kaplan, D. (1988). The impact of specification error on the estimation, testing, and improvement of structural equation models. *Multivariate Behavioral Research*, 23(1), 69–86.
- Keller, B. T., & Enders, C. K. (2019). *Blimp user's guide (version 2.0)*. Los Angeles, CA.
- Keller, B. T., Enders, C. K., & Kim, D. S. (2019). Three-level multiple imputation: A fully conditional specification approach. *Manuscript in preparation*..
- Kim, S., Belin, T. R., & Sugar, C. A. (2018). Multiple imputation with non-additively related variables: Joint-modeling and approximations. *Statistical methods in medical research*, 27(6), 1683–1694.
- Kim, S., Sugar, C. A., & Belin, T. R. (2015). Evaluating model-based imputation methods for missing covariates in regression models with interactions. *Statistics in medicine*,

- 34(11), 1876–1888.
- Kreft, I., & de Leeuw, J. (1998). *Introducing multilevel modeling*. Newbury Park, CA: Sage.
- Little, R. J., & Rubin, D. B. (2002). *Statistical analysis with missing data*. Wiley.
- Liu, H., Zhang, Z., & Grimm, K. J. (2016). Comparison of inverse wishart and separation-strategy priors for bayesian estimation of covariance parameter matrix in growth curve analysis. *Structural Equation Modeling: A Multidisciplinary Journal*, 23(3), 354–367.
- Lynch, S. M. (2007). *Introduction to applied bayesian statistics and estimation for social scientists*. Springer Science & Business Media.
- Maas, C., & Hox, J. (2005). Sufficient sample sizes for multilevel modeling. *Methodology*, 1, 86-92. doi: 10.1027/1614-1881.1.3.86
- Mistler, S. A., & Enders, C. K. (2017). A comparison of joint model and fully conditional specification imputation for multilevel missing data. *Journal of Educational and Behavioral Statistics*, 42(4), 432–466.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Murray, D. M., & Blitstein, J. L. (2003). Methods to reduce the impact of intraclass correlation in group-randomized trials. *Evaluation Review*, 27(1), 79–103.
- Plummer, M. (2016). *Jags: A program for analysis of bayesian graphical models using gibbs sampling (version 4.2.0)*. Retrieved from [Retrievedfromhttp://sourceforge.net/projects/mcmc-jags/](http://sourceforge.net/projects/mcmc-jags/)
- Preacher, K. J., Zyphur, M. J., & Zhang, Z. (2010). A general multilevel sem framework for assessing multilevel mediation. *Psychological methods*, 15(3), 209.
- R Core Team. (2018). *R: A language and environment for statistical computing*. Vienna, Austria. Retrieved from <https://www.R-project.org/>
- Raudenbush, S. W., & Bryk, A. S. (2002). *Hierarchical linear models: Applications and data analysis methods*. Sage.
- Rights, J. D., & Sterba, S. K. (2018). Quantifying explained variance in multilevel models: An integrative framework for defining r-squared measures. *Psychological methods*.
- Ripley, B. D. (1987). *Stochastic simulation* (Vol. 5). Wiley Online Library.
- Royston, P. (2005). Multiple imputation of missing values: update. *The Stata Journal*, 5(2),

188–201.

- Rubin, D. B. (1976). Inference and missing data. *Biometrika*, *63*(3), 581–592.
- Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys*. John Wiley & Sons.
- Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. Chapman and Hall/CRC.
- Seaman, S. R., Bartlett, J. W., & White, I. R. (2012). Multiple imputation of missing covariates with non-linear effects and interactions: an evaluation of statistical methods. *BMC medical research methodology*, *12*(1), 46.
- Snijders, T. A., & Bosker, R. J. (2012). *Multilevel analysis: An introduction to basic and advanced multilevel modeling* (2nd ed.). Thousand Oaks, CA: Sage.
- Van Buuren, S. (2007). Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical methods in medical research*, *16*(3), 219–242.
- Van Buuren, S. (2012). *Flexible imputation of missing data*. Chapman and Hall/CRC.
- Van Buuren, S., Brand, J. P., Groothuis-Oudshoorn, C. G., & Rubin, D. B. (2006). Fully conditional specification in multivariate imputation. *Journal of statistical computation and simulation*, *76*(12), 1049–1064.
- von Hippel, P. T. (2009). How to impute interactions, squares, and other transformed variables. *Sociological methodology*, *39*(1), 265–291.
- White, I. R., Royston, P., & Wood, A. M. (2011). Multiple imputation using chained equations: issues and guidance for practice. *Statistics in medicine*, *30*(4), 377–399.
- Yucel, R. M. (2008). Multiple imputation inference for multivariate multilevel continuous data with ignorable non-response. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1874), 2389–2403.
- Zhang, Q., & Wang, L. (2017). Moderation analysis with missing data in the predictors. *Psychological Methods*, *22*(4), 649.