Informative Variance Priors for Bayesian Multilevel Models

Liu Liu

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2024

Reading Committee:

Elizabeth A. Sanders, Chair

Min Li

Chun Wang

Program Authorized to Offer Degree:

College of Education

© Copyright 2024

Liu Liu

University of Washington

Abstract

Informative Variance Priors for Bayesian Multilevel Models

Liu Liu

Chair of the Supervisory Committee:

Elizabeth A. Sanders

College of Education

This dissertation is focused on the use of informative variance component priors for small- and large-sample Bayesian multilevel modeling. Specifically, I explored the transformative relationships among chi-square, gamma, and inverse-gamma distributions to develop tailored informative priors that may be useful in modeling data with small L2 sample sizes. First, I used Monte Carlo simulation to compare model parameter estimation with maximum likelihood and Bayesian MCMC uninformative, weak informative, and strong informative priors across design conditions including different numbers of clusters (J = 10, 30, 100), cluster sizes (M = 5, 30), intraclass correlations (ICCs; .01, .20, and .40), and levels of explained variance at L1 and L2 (R-squared = .0, .2, .4). The simulation results indicated that the use of strongly informative variance component priors (i.e., inverse-gamma with hyperparameters set to have a mode = true variance) was better than Uniform (flat/uninformative) variance priors, and was also better than

weak informative priors (i.e., inverse-gamma with shape and scale hyperparameters set to .01 and .01, respectively) when dealing with variables measured on typical education scales (e.g., standard deviations of 100). In addition to simulations, I also demonstrated the use of variance priors to analyze the TIMSS 2019 Grade 8 U.S. Science achievement data, with both the full sample (N = 273 schools) as well as a randomly selected small-sample subset (n = 30 schools). Those results were consistent with the simulation findings but did in fact show that the use of strongly informative priors was best: it captured the full sample's variance but with the advantage of a smaller credible interval width. Taken together, these findings suggest that, in real-world settings, use of an informative variance prior is advantageous because it will more precisely capture the true variability in L2 units (the context in which we measure individuals), particularly when the number of L2 units is small.

DEDICATION

To my beloved parents

ACKNOWLEDGEMENTS

I would like to express my gratitude to many who inspired, supported, and accompanied me throughout my PhD journey. I owe a huge debt of gratitude to all who reviewed early drafts of this writing and provided feedback and encouragement. Foremost, I extend heartfelt thanks to my committee members: Drs. Elizabeth A. Sanders, Min Li, Chun Wang, Julia Duncheon, and Valerie Manusov. I also thank Dr. Adrian Dobra for his service on my committee during the general exam process. This dissertation would not have been possible without their suggestions and support.

Special thanks to my thesis chair and advisor, Dr. Elizabeth A. Sanders, for her invaluable guidance and support. Her meticulous attention to detail and dedication ensured that all analyses were set up properly and guide me through the thesis process. I deeply appreciate all the conversations we had, all the projects we collaborated on, and the walking we had together. I feel so fortunate to be her student and to have learned from her passion and work ethic. Dr. Sanders not only provided academic support but also offered insightful advice on personal conduct and how to better integrate into American culture, greatly enriching my personal and professional growth.

I am also grateful to Dr. Min Li, who consistently encouraged me to explore new research areas. Her knowledge and guidance were essential to the completion of this dissertation and have taught me valuable lessons about life and research. Her belief in my capabilities fueled my determination during challenging times. She also imparted wisdom on life, which has greatly helped my personal growth.

I am also thankful to Dr. Chun Wang for her insightful feedback on my dissertation. Her detailed critiques and constructive suggestions enhanced the quality of my work.

The journey through my doctoral studies coincided with the unprecedented challenges of a global pandemic, which tested my mental and physical resilience. I am grateful for the support of my family and friends, whose love and encouragement have been both a comfort and a strength to me.

Moreover, I would like to acknowledge my doctoral committee, the Measurement & Statistics program, and the University of Washington. Their collective efforts have provided an enriching and great environment that has motivated my scholarly pursuits.

I would also like to acknowledge the use of ChatGPT as a language editor to correct grammar and improve clarity in my writing. The citation for the tool is provided in the References section. Importantly, while the tool assisted with language editing, all content ideas and the synthesis of others' ideas were solely the product of my own effort.

As I approach the final stages of my doctorate, I am filled with mixed emotions. Although I defended in June, truly concluding my doctoral journey brings a deep sense of reluctance to let go.

Finally, my gratitude extends to all those who contributed directly or indirectly to my academic journey. The collective wisdom, support, and kindness I have received have left a pivotal mark on my personal and professional life.

TABLE OF CONTENTS

LIST OF TABLES vi
LIST OF FIGURES viii
LIST OF ABBREVIATIONS xi
CHAPTER ONE: INTRODUCTION 1
CHAPTER TWO: BAYESIAN STATISTICS 3
Differences between Frequentist and Bayesian Framework 5
Bayesian Analysis Prior Selection 9
CHAPTER THREE: BAYESIAN MULTILEVEL MODELING 11
Brief Overview of Multilevel Modeling 11
Bayesian MLM Prior Distributions for Variance Components 14
Relationships among Distribution Families 18
Non-Transformable Distributions 29
CHAPTER FOUR: SOFTWARE FOR BAYESIAN ANALYSIS 34
CHAPTER FIVE: DEFAULT VARIANCE PRIORS IN BAYESIAN MLM SOFTWARE 38
R Packages 38
Other Popular Software 39
CHAPTER SIX: RESEARCH ON BAYESIAN MLM VARIANCE PRIORS 40
CHAPTER SEVEN: REVIEW OF APPLIED RESEARCH USING BAYESIAN ANALYSIS 43
CHAPTER EIGHT: MONTE CARLO SIMULATION STUDY 46
Data Generation 46
Methods 48
Results 49
CHAPTER NINE: REAL DATA APPLICATION EXAMPLE 52
Research Context and Dataset Description 52
Methods 53
Results 59
CHAPTER TEN: EQUIVALENCIES AMONG INFORMATIVE VARIANCE PRIORS USING BRMS 61
Methods 61
Results 64
CHAPTER ELEVEN: DISCUSSION 65

Limitations and Conclusion 67
REFERENCES 69
APPENDIX 146
1. Example Mplus code for Simulation Data Generation (J = 10, m = 5, ICC = .40) 146
2. Example Mplus code for Analyses with Uninformative Priors (J = 10, m = 5, ICC = .40) 146
3. Example Mplus code for Analyzing Datasets with Weak Informative Normal Coefficient Priors and Strong Informative Inverse Gamma Variance Priors (J = 10, m = 5, ICC = .40).. 147
4. Example R code for Real Data Analysis 147

LIST OF TABLES

Table 1 Similarities and Differences between Frequentist and Bayesian Statistics 90

Table 2 Overview of the Common Probability Distributions and Their Priors 91

Table 3 Relationships among Families of Distributions Commonly used as Priors in Bayesian Analyses 92

Table 4 Overview of Key Software Tools for Bayesian Statistical Analysis 93

Table 5 Supported Bayesian Multilevel Models by R Packages 95

Table 6 Overview of Previous Research on Bayesian Variance Priors 96

Table 7 False Positive Rate and Correct Positive for L1 Predictor Slope Coefficient, Collapsed across L2 Predictor Effects 97

Table 8 False Positive Rates for L2 Intercept Coefficient and Raw Bias for L2 Intercept and Variance 98

Table 9 False Positive Rate and Correct Positive for L2 Predictor Slope Coefficient, Collapsed across L1 Predictor Effects 99

Table 10 Credible Interval Coverage Results for Variance Components for Conditions with J = 30 100

Table 11 Estimation Time for the Bayesian Analyses Focusing on Inverse Gamma Prior Informativeness for TIMSS Data 101

Table 12 Steps for Calculating Inverse Gamma Prior Informativeness using Full Sample for Bayesian Multilevel Analysis 102

Table 13 Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Inverse Gamma Prior Informativeness (Full Sample) 103

Table 14 Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Inverse Gamma Prior Informativeness (Subsample)..... 104

Table 15 Estimation Time for the Bayesian Analyses Focusing on G, IG, Half-*t*, and Half-Cauchy Informative Priors for TIMSS Data..... 105

Table 16 Steps for Calculating Strongly Informative Half-*t* and Half-Cauchy Priors using Full Sample for Bayesian Multilevel Analysis..... 106

Table 17 Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Informative G, IG, Half-*t*, and Half-Cauchy (Full Sample)..... 108

Table 18 Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Informative G, IG, Half-*t*, and Half-Cauchy (Subsample)..... 109

LIST OF FIGURES

Figure 1 Comparison of Regression Parameter Estimates with Frequentist and Bayesian Approaches 110

Figure 2 Examples of the Chi-Square Distribution Family with Varying Degrees of Freedom 111

Figure 3 Examples of the Gamma Family of Distributions with Varying Hyperparameters 112

Figure 4 Examples of the Inverse Gamma Family of Distributions with Varying Hyperparameters 113

Figure 5 Examples of the Normal Family of Distributions with Varying Hyperparameters..... 114

Figure 6 Examples of the Half-t Family of Distributions with Varying Hyperparameters 115

Figure 7 Examples of the Half-Cauchy Family of Distributions with Varying Hyperparameters 116

Figure 8 Examples of the Uniform Family of Distributions with Varying Hyperparameters ... 117

Figure 9 False Positive Rates for L2 Intercept Coefficient by Design Condition, Collapsed across L1 Predictor Effects 118

Figure 10 Raw Bias Results for L2 Intercept Coefficient by Design Condition, Collapsed across L1 Predictor Effects 119

Figure 11 Raw Bias Results for L2 Intercept Residual Variance by Design Condition, Collapsed across L1 Predictor Effects 120

Figure 12 False Positive Type I Error Rates and Correct Positive Power for L2 Predictor Slope Coefficient by Design Condition, Collapsed across L1 Predictor Effects 121

Figure 13 Raw Bias Results for L2 Predictor Coefficient by Design Condition, Collapsed across L1 Predictor Effects 122

Figure 14 L2 Variance Estimate Bias for N = 20 replications for Scaled Data with True SD = 100 123

Figure 15 Chi-square, Gamma and Inverse Gamma Distributions with Full Sample 124

Figure 16 Chi-square, Gamma and Inverse Gamma Distributions with Subsample 125

Figure 17 Estimated L2 Intercept Standard Deviation Using Inverse Gamma Prior Informativeness (Full Sample) 126

Figure 18 Estimated L2 Intercept Standard Deviation Using Inverse Gamma Prior Informativeness (Subsample) 127

Figure 19 Estimating L1 Intercept Using Inverse Gamma Prior Informativeness (Full Sample) 128

Figure 20 Estimating L1 Math (Centered, *Z*) Effects Using Inverse Gamma Prior Informativeness (Full Sample) 129

Figure 21 Estimating L2 Math (Aggregate, *Z*) Effects Using Inverse Gamma Prior Informativeness (Full Sample) 130

Figure 22 Estimating Interaction Effects of L1 and L2 Math Using Inverse Gamma Prior Informativeness (Full Sample) 131

Figure 23 Estimating L1 Intercept Using Inverse Gamma Prior Informativeness (Subsample) 132

Figure 24 Estimating L1 Math (Centered, *Z*) Effects Using Inverse Gamma Prior Informativeness (Subsample) 133

Figure 25 Estimating L2 Math (Aggregate, *Z*) Effects Using Inverse Gamma Prior Informativeness (Subsample) 134

Figure 26 Estimating Interaction Effects of L1 and L2 Math Using Inverse Gamma Prior Informativeness (Subsample) 135

Figure 27 Estimated L2 Intercept Standard Deviation Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Full Sample) 136

Figure 28 Estimated L2 Intercept Standard Deviation Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Subsample) 137

Figure 29 Estimating L1 Intercept Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Full Sample) 138

Figure 30 Estimating L1 Math (Centered, *Z*) Effects Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Full Sample) 139

Figure 31 Estimating L2 Math (Aggregate, *Z*) Effects Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Full Sample) 140

Figure 32 Estimating Interaction Effects of L1 and L2 Math Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Full Sample) 141

Figure 33 Estimating L1 Intercept Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Subsample) 142

Figure 34 Estimating L1 Math (Centered, *Z*) Effects Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Subsample) 143

Figure 35 Estimating L2 Math (Aggregate, *Z*) Effects Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Subsample) 144

Figure 36 Estimating Interaction Effects of L1 and L2 Math Using Informative Gamma, Inverse Gamma, Half-*t*, and Half-Cauchy (Subsample) 145

LIST OF ABBREVIATIONS

ADMB	Automatic Differentiation Model Builder
BUGS	Bayesian Inference Using Gibbs Sampling
CDF	Cumulative Distribution Function
Chi	Chi-square
CI	Confidence Interval
CLT	Central Limit Theorem
CrI	Credible Interval
G	Gamma
HMC	Hamilton Monte Carlo
ICC	Intraclass Correlation
IG	Inverse Gamma
JAGS	Just Another Gibbs Sampler
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
MLM	Multilevel Model
N	Normal Distribution
NUTS	No-U-Turn Sampler
PDF	Probability Density Function
PSRF	Potential Scale Reduction Factor
REML	Restricted Maximum Likelihood
SEM	Structural Equation Modeling
SRMR	Standardized Root Mean Square Residual
TIMSS	Trends in International Mathematics and Science Study
U	Uniform Distribution

CHAPTER ONE:

INTRODUCTION

Accurately modeling variance component parameters in multilevel models (MLMs) with frequentist maximum likelihood (ML) estimation can be difficult when the number of higher-level units is small or when the model random factor covariance structure is complex. In these cases, Bayesian model estimation has the potential to provide comparatively more accurate variance component estimates, along with more precise coefficient credible intervals, particularly when more informative (and accurate) priors are employed. As is well known, a Bayesian analysis approach requires selection of priors for model coefficients as well as variance components – yet choosing priors, especially informative priors, for variance components can be more complex than choosing coefficient priors. This is in part because statistical software packages can differ in default priors used, especially for variance components, and also because there is yet to be consistent advice for choosing from among available prior distributions (e.g., Baldwin & Fellingham, 2013; Browne & Draper, 2006; Gelman, 2006; McNeish, 2016a).

This dissertation used Monte Carlo simulation to evaluate the impact of different prior distribution and hyperparameter selections on parameter accuracy and precision for 2-level hierarchical regression models. Appropriate choices for scalar variance component prior distributions include the chi-square (*Chi*), Gamma (*G*) and Inverse-Gamma (*IG*), Gaussian/normal (*N*), folded/half-*t* and half-Cauchy (a special case of half-*t*), and Uniform (*U*) distributions (e.g., Baek et al., 2020; Gelman, 2006; McNeish, 2016a; Moeyaert et al., 2017). However, I focused mainly on the *IG* prior in simulations since this distribution is widely available across numerous software platforms. My focal aims were as follows.

1. Evaluate the **impact of informativeness levels of different scalar variance priors** in analyzing 2-level simulated data with varying sample sizes, intraclass correlations (ICCs), and predictor effect sizes.
2. Provide a real-data **demonstration** of the use of many different Bayesian informative variance priors when analyzing 2-level data from subsamples of the TIMSS 2019 Grade 8 U.S. Science achievement data.
3. Provide a **table of recommendations** for informative variance component priors with a focus on those used in the simulations and real data demonstration.

CHAPTER TWO:

BAYESIAN STATISTICS

Frequentist model estimation methods rely solely on data available and attempt to estimate model parameters that are considered to be constant albeit unknown. In this framework, researchers are concerned with the level of the probability of observing their results when the null hypothesis is true (e.g., Shreffler & Huecker, 2022; Vasishth & Nicenboim, 2016, p. 592). The null hypothesis claims that there is no association between a given input variable (also known as independent variables, predictor variables, and explanatory variables, to name a few) and a given output variable (also known as dependent variables, outcome variables, and response variables). If the null hypothesis is true, and the mathematical probability of observing the values in the sample turns out to be 0.05 or less, the researcher considers the result to be sufficiently rare and as evidence that the null may not be true after all. However, it is possible of course to observe rare values by chance and so the risk of falsely rejecting the null always exists.

Bayesian model estimation is grounded in Bayes' theorem (also known as Bayes' rule). It computes the probability of an observed value occurring based on prior knowledge of conditions that might be related to the value (Gelman, 1995; Gelman et al. 2014; Kruschke, 2014; Levy & McNeish, 2021; Stuart & Ord, 2010) and does not assume any type of null hypothesis. The formula is expressed as follows.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1)$$

Where:

- $P(A|B)$ is the conditional probability of A given B, representing the probability of event A occurring given that event B has occurred.

- $P(B|A)$ is the conditional probability of B given A, representing the probability of B occurring given that A has occurred.
- $P(A)$ is the prior probability of A, representing the probability of A occurring without any knowledge of B.
- $P(B)$ is the prior probability of B, representing the probability of B occurring without any knowledge of A.

Three ingredients comprise the Bayesian framework to model estimation: the prior, the likelihood, and the posterior. All three components are probability distributions. The *prior* represents the assumptions about the truth of the values in the population; the *likelihood* represents the observed data (sample); and the *posterior* represents the integration of the prior with the likelihood (van de Schoot et al., 2014). That is, the posterior tells us how our belief (prior) should be updated (posterior), given the data (likelihood).

For any Bayesian analysis, all unknown model parameters must be assigned with a prior. By computing an *empirical* posterior distribution, Bayesian estimation can potentially overcome sample-specific biases that a frequentist model estimation approach may have (Finch et al., 2019; Ghosh & Albert, 2009). Although simple models can be estimated analytically, most Bayesian analyses require Markov Chain Monte Carlo (MCMC) iterative sampling to estimate the posterior distribution of model parameters (Carlin & Chib, 1995; Van Ravenzwaaij et al., 2018). The iterative sampling method is similar to Maximum Likelihood in that parameter values are sampled iteratively, but in Bayesian analysis, the iterations are for the posterior distribution, and these iterations can be re-run more than once (defined by the number of chains). The first set of iterations of each chain are usually discarded as “warm-ups” or “burn-in” as they can be sensitive to random starting values (e.g., Finch et al., 2019). Finally, in theory the MCMC

process can continue infinitely and so a stopping rule for convergence is required (e.g., *r*-hat values reaching a certain threshold); as well, satisfactory convergence can be assessed by inspecting trace plots (much like recordings during an earthquake).

Differences between Frequentist and Bayesian Framework

The main differences between frequentist and Bayesian model estimation include the nature of the probability associated with a given model parameter, and the nature of the unknown parameters to be estimated. Table 1 (adapted from van de Schoot et al., 2014) highlights these and other differences among the two model estimation methods; below I provide further description of these elements.

Probability

The frequentist framework for statistical analysis (which has served as the conventional framework in the social sciences for many decades) associates the probability of observations with *theoretical* mathematically derived frequencies (van de Schoot et al., 2014). A simple example is repeatedly flipping a coin and focusing on a "tails" landing. All possible results (heads and tails) comprised the sample space, and the probability is the proportion of a specific outcome (e.g., tails) relative to the number of times the coin is flipped. Similarly, conclusions about the observed results of an education experiment assume that, under the same conditions, the educational experiment could be conducted many (infinite) times to attain a repeatable result. In contrast, the Bayesian perspective views probability as associated with a combination of subjective experience (priors) and the observed data (likelihood). In other words, when we make a guess about something, the guess involves using any prior information that we can recall; when we find out the true answer about our guess, we can update our "priors" accordingly.

In education and other behavioral and social sciences, a relevant question could be, "based on past research, what is the probability that the intervention proposed in the study will have an effect greater than 10 points over the control group?" (Kaplan, 2019, p. 1). The probability, in this case, is based on how much prior knowledge researchers have about similar interventions in the previous research studies and how accurate they think their prior knowledge is, rather than relying on an infinite number of repeatable and independent studies as that of the frequentist approach (Wasserman, 2004).

Unknown Parameters

The nature of unknown parameters is another difference between the frequentist and Bayesian approaches. For example, the unknown parameters in education policy research may concern the true impact of an intervention. In the frequentist view, the treatment effect to be estimated is unknown but theoretically has an exact *fixed* value. In contrast, the Bayesian views unknown parameters as *randomly* varying. The posterior probability distribution describes this probable range of the treatment effect based on the researchers' background knowledge about the treatment effect, their uncertainty about that background knowledge, and the observed data (e.g., Jackman, 2009; Kaplan, 2023).

Hypothesis Testing

With respect to hypothesis testing, frequentist and Bayesian frameworks offer very different views. As previously mentioned, frequentist model estimation methods usually involve testing the null hypothesis that there is no relation among variables of interest – in other words, that "nothing happened." In short, the null is assumed true, and the job of the researcher is to disprove the null based on an arbitrary (but traditional) decision rule such as a probability of .05 or less. However, the null hypothesis may not be a good starting point because we almost always

expect that "something will happen" (van de Schoot et al., 2014). The Bayesian approach does not assume the null is true to evaluate model results. Instead, it focuses on the question of "what is the probable location of the parameter of interest (e.g., treatment effect), given the observed data and my prior knowledge?"

Added Questions

In both frequentist and Bayesian frameworks, researchers can evaluate point estimates and parameter intervals (confidence intervals within the frequentist approach and credible intervals within the Bayesian approach). However, more questions can be evaluated using a Bayesian framework, such as: "How credible are our assumptions given the data?" And additionally, "What is the probability that the intervention effect is between the X and Z values?" where X and Z may be the values related to the education policy evaluation (Kaplan, 2019, p. 2).

Results Interpretability

Another difference between frequentist and Bayesian statistics, especially for scholars who conduct applied research, concerns results interpretation. The posterior distribution of the Bayesian model can be directly interpreted as the estimation of unknown parameters, and the uncertainty of this estimation is shown in the range of distribution. The confidence interval (CI) of the frequentist method is usually based on the assumption of normality that is defined as symmetrical (e.g., point estimation ±1.96 SE), which the CI of random effects in the frequentist method can include negative values (Ruf et al., 2021). However, random effects are expressed in variance or standard deviation, which cannot be negative. The Bayesian approach avoids this issue by incorporating a prior distribution with only non-negative values; thus, the posterior distribution does not include a negative estimation of random effect variation. In addition, *the interval estimates are interpreted differently*. Specifically, the 95% CI of the frequentist

estimation methods is interpreted as "the parameter value of interest is within this XYZ range, with 95% confidence" (because 95% of intervals will contain the true population value), whereas the Bayesian interpretation for the credible interval (sometimes denoted CrI to differentiate it with the frequentist CI) is more direct: "95% of the parameter values of interest are contained in this interval" (e.g., van de Schoot et al., 2021).

Model Estimation

Ultimately, the key difference between frequentist and Bayesian analysis methods that underpins all the previously discussed elements is *how model parameters are estimated*. In the frequentist framework, model parameters usually are typically estimated using ordinary least squares (OLS) or a Maximum likelihood (ML) algorithm, both of which rely on large-sample sampling theory with sample size requirements that are not absolute but dependent on the size and complexity of the model (Hox & McNeish, 2020). In the case of ML, model estimates are derived by maximizing the probability (or minimizing the log of the likelihood) that the observed data come from the specified model. In this process, variance estimates (which are an important part of multilevel models) are computed by assuming the parameter estimates are the true parameter values, which underestimates variance in smaller samples, in turn inflating Type I error (Hox & McNeish, 2020). Restricted Maximum Likelihood (REML) and OLS can minimize this bias; however, REML cannot handle missingness like ML can, and it is less likely to converge on a solution for complex models (Baek & Ferron, 2020). Further, REML does not allow researchers to compare results across nested models.

A second concern in the literature around modeling small-sample data using ML or REML is that model assumptions, such as normally distributed residuals, can be untenable, and even if they are tenable, there will naturally be a lack of power to detect true effects. One rule of

thumb for structural equation models using ML estimation is that a minimum of 100 or 200 cases are needed (Boomsma, 1982, 1985). In other words, modeling multilevel data using a frequentist approach may be limited due to concerns about estimating level 2 (L2) variance with small L2 sample sizes in terms of potential underestimation (bias) as well as lack of precision.

As previously alluded to, Bayesian model estimation does not rely on large-sample theory. Like coefficient model parameters, the Bayesian estimation process will provide variance posterior distributions that take into account the prior used and the observed data (e.g., Gelman et al., 2014; McNeish, 2016a,b). In this way, Bayesian estimation overcomes potential bias and imprecision that can occur using ML estimation (Finch et al., 2019; Ghosh & Albert, 2009).

Bayesian Analysis Prior Selection

As discussed above, in a Bayesian analysis, the posterior distribution provides parameter estimates that are a product of our priors and the observed data. A prior distribution for the parameter of interest must be specified in advance by the researcher based on their beliefs and knowledge about the shape of the true parameter distribution, which can take one of two forms: uninformative or informative. *Uninformative* priors are not based on prior research and are used when there is little or no information known about a parameter in advance. Thus, an uninformative prior only serves to produce a posterior distribution but has no effect on the results beyond the observed data and will produce estimates similar or even identical to frequentist estimates (e.g., Lemoine, 2019; McNeish, 2016a; Rupp et al., 2004).

Informative priors, on the other hand, are used when reasonable information about the parameters is known (e.g., Gelman, 2006). In addition, this type of prior can range from weak to strong depending on the level of informativeness of the prior. Figure 1 illustrates this idea by showing how a hypothetical normally distributed slope's sampling distribution would be similar

across frequentist (leftmost panel) and Bayesian analyses when the Bayesian analysis invokes a flat, non-informative prior (middle panel), whereas an informative prior yields a posterior distribution that combines information from the data and the prior (rightmost panel).

One uninformative prior a researcher could use includes the uniform (rectangular) distribution centered at 0 with a very wide range, such as 10,000 (i.e., there are equal probabilities of any value between -10,000 and +10,000). Alternatively, using the normal distribution with a mean of 0 and variance of 10,000 might also be used, but this is slightly more informative than the uniform distribution because the probabilities of values decrease as values move away from the center of the normal curve. Yet, in both cases, use of a very wide range of values for a prior distribution lacks any reliable information for use in the parameter estimation process, leaving the posterior distribution to depend on the observed data distribution (likelihood). In contrast, an informative prior might use the normal distribution with a specific mean and variance that is reflected in the observed data or from prior research.

In addition to selecting a prior based on previous information about the parameter of interest, *small samples* require even more careful thought. This is because the prior will have a far greater impact the posterior distribution for small samples because the data offer little information and will be downweighted in the integration between the prior and the data. Use of a strong informative prior, on the one hand, could lead to a biased posterior if the prior is biased. But a less obvious issue is that uninformative (e.g., flat) priors can also result in bias if the algorithm samples an extreme value from the prior (McNeish, 2016a,b; Smid et al., 2020b).

CHAPTER THREE:

BAYESIAN MULTILEVEL MODELING

Brief Overview of Multilevel Modeling

Multilevel models (MLMs), also known as hierarchical, random effects, and mixed effect models, are an extension of classical unilevel[1] regression models and have become ubiquitous in the social and behavioral sciences (e.g., Hox & Roberts, 2011; Luo et al., 2021; McNeish et al., 2019). These models incorporate dependencies in cross-sectional data, such as students (level 1, L1) within schools (level 2, L2), as well as in repeated measures data, such as multiple measurements (L1) of persons (L2) over time. A standard hierarchical 2-level population MLM is as follows (e.g., Snijders & Bosker, 2011, p. 96):

$$Y_{ij} = \gamma_{00} + \gamma_{10}X_{ij} + U_{0j} + U_{1j}X_{ij} + \varepsilon_{ij}, \quad (2)$$

where Y_{ij} is the dependent variable for individual i in cluster[2] j; γ_{00} is the intercept (the expected value of Y when X = 0, for clusters with average intercept values); γ_{10} is the fixed effect of the predictor X, also known as the slope coefficient (the expected change in Y for each one-unit increase in X, for clusters with average intercept and slope values); X_{ij} is the cluster-mean centered value of the predictor for individual i in cluster j; U_{0j} is the random intercept deviation for cluster j (how much the intercept for cluster j deviates from γ_{00}); $U_{1j}$is the random slope deviation for cluster j (how much the predictor slope for cluster j deviates from γ_{10}); and ε_{ij} is the residual error term for individual i in cluster j (how much the individual's dependent variable value deviates from their model-implied value).

[1] Some authors use the term 'single level' but I adopt unilevel for brevity.
[2] I use the term 'cluster' for *J* Level 2 units rather than 'group' to avoid confusion with a fixed effect group predictor such as demographic groupings.

The random effects U_{0j} and U_{1j} are assumed to be multivariate normally distributed with mean 0, variances τ_{00} and τ_{11}, respectively, and a covariance τ_{01}. The residual errors ε_{ij} are assumed to be independently and identically normally distributed with mean 0 and variance σ^2; it is also assumed that ε_{ij} are independent of U_{0j} and U_{1j}. This model can flexibly accommodate additional predictors at L1 or L2, as well as have constraints placed on the random factor variances and covariances.

One challenge in MLM estimation pertains to situations where the higher-level units (here, at L2) are limited in size—which occurs in research with limited budgets or data collection time frames, as well as longitudinal studies following rarer populations such as young children with Autism. In these circumstances, estimating variance components (e.g., τ_{00} and τ_{11}) that assume the random effects (e.g., U_{0j} and U_{1j}, respectively) are normally distributed can be problematic (Heck & Thomas, 2020; McNeish & Stapleton, 2016a,b). Of course, MLM alternatives for clustered data exist when variance component assumptions using frequentist estimation are untenable. One popular approach is to treat J clusters as a set of dummy- or effect-coded $J-1$ individual-level predictors, and model the data using unilevel "fixed effects" regression. However, this approach precludes using L2 predictors as they would be collinear with cluster membership and cannot handle more than two levels in the data structure. Yet other MLM alternative approaches include using unilevel regression with cluster-robust standard errors or employing generalized estimating equations.

An arguably better approach to MLM alternatives for modeling clustered data with relatively small numbers of clusters, particularly if L2 predictor effects are of substantive interest or if there are more than two levels in the hierarchy, is to specify an MLM in which parameters are estimated within a Bayesian framework. As discussed previously, in contrast to a frequentist

approach that invokes large-sample theory about the behavior of cluster deviations in the intercept and/or slope (i.e., the U_{0j} and/or U_{1j} values) to estimate model parameters, a Bayesian analysis incorporates "prior" probability distributions that are then integrated with the observed data in order to arrive at a "posterior" probability distribution of parameter values (e.g.., Gelman et al., 2014; Kruschke, 2014). In short, a Bayesian framework constructs "credible intervals" based on an observed middle percentage of the posterior distribution, which can be any distributional shape (not necessarily normal).

While selecting a prior distribution with related hyperparameters (which I collectively term as a "prior") for linear and generalized linear regression fixed effect coefficients is generally easy (typically a normal distribution with an assumed mean and standard deviation based on descriptive statistics or previous research), choosing appropriate priors for MLM variance components, especially *informative* priors, is less intuitive. Specifically, variance prior distributions must contain non-negative values and follow an appropriately right-skewed shape, and prior distribution options (and defaults) vary across software packages. In the following section, I describe the prior distributions most commonly used for variance components in Bayesian analyses, including the chi-square (*Chi*), Gamma (*G*) and Inverse-Gamma (*IG*), Gaussian/normal (*N*), folded/half-*t* and half-Cauchy (a special case of half-*t*), and Uniform (*U*) distributions (e.g., Baek et al., 2020; Gelman, 2006; Gelman et al., 2014; McNeish, 2016a; Moeyaert et al., 2017). After describing each type of distribution in isolation, I highlight their relationships for potential transformations among hyperparameters to create similarly scaled informative prior distributions.

Bayesian MLM Prior Distributions for Variance Components

Chi-Square. The Chi-square distribution (χ^2 or Chi-square) is a family of continuous probability distributions that are a special case of the gamma family (described next). Specifically, it is the sampling distribution for the sum of the squared deviations among v independent standard normal random variables (Bhaumik et al., 2009), where v is the degrees of freedom governing its shape (see Figure 2). The probability density function (PDF) of the chi-square distribution can be expressed as:

$$f(x;v) = \begin{Bmatrix} \frac{x^{v/2-1}e^{-x/2}}{2^{v/2}\Gamma(v/2)}, & x > 0; \\ 0, \quad \text{otherwise} \end{Bmatrix} \tag{3}$$

where $\Gamma(v/2)$ is the gamma function (Kissell & Poserina, 2017).

The mean and variance of the chi-square are v and $2v$, respectively. The smaller the degrees of freedom, the more right-skewed the distribution is (see Figure 2). In large-sample theory, the variances of random normal variables are distributed as chi-square. Thus, the chi-square is a natural potential prior distribution for variance components.

Gamma. The Gamma (Γ or *G*) distribution family is a 2-parameter family of continuous probability distributions that are right-skewed, characterized by a shape parameter α (or *a*) and a rate parameter β (or *b*). It is often used as a prior distribution in Bayesian statistical analyses when the parameter of interest must be positive and is likely to be skewed, such as the variance in the dependent variable due to clustering in a 2-level MLM. The PDF of the gamma distribution is given by:

$$f(x;\ \alpha,\beta) = \frac{\beta^{\alpha}x^{\alpha-1}e^{-\beta x}}{\Gamma(\alpha)} \tag{4}$$

for $x > 0$, $\alpha > 0$, $\beta > 0$, where α is the shape parameter, β is the rate parameter, and $\Gamma(\alpha)$ is the gamma function evaluated at α (Johnson et al., 1995; Kaplan, 2023; Ulloa, 2018). The mean and

variance of the gamma are α/β and α/β^2, respectively. Importantly, shape parameter primarily affects the gamma's symmetry and skewness, making it more symmetric as α increases. The rate parameter influences how concentrated the gamma is around its mean, with higher β values leading to a more narrowly concentrated distribution.

The mean of the gamma increases as the shape parameter increases, which means with higher α values, the average value of the distribution shifts to the right (see Figure 3). In Bayesian analysis, gamma is often used as a default for variance component priors (e.g., Gelman, 2006; Gelman et al., 2014) because it is a conjugate prior to the precision parameter in a normal distribution (i.e., the posterior distribution is also a gamma distribution). In this context, a "conjugate prior" is an initial probability assumption, resulting posterior distribution having the same distribution form as the prior. For example, if both the likelihood and the prior are normal distributions, the posterior distribution will also be normal.

Inverse Gamma. Inverse gamma (*IG*) family of distributions is also a conjugate prior distribution in Bayesian analyses. Like the gamma, the *IG* is positive and right skewed and has a shape α and a scale parameter β; however, it is more concentrated near zero (see Figure 4, upper panel). The *IG* is often used as an informative prior because it can express prior information about the expected range and variability of the variance component (Gelman, 2006; Llera & Beckmann, 2016). As Gelman (2006) proposed, the *IG* with shape and scale parameters of 0.01 and 0.01 is one of the most popular weak prior for variance components (see Figure 4 lower panel). Its PDF is given by:

$$f(x;\ \alpha,\ \beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{-\alpha-1} e^{-\frac{\beta}{x}} \qquad (5)$$

for $x > 0, \alpha > 0\ , \beta > 0, \Gamma(\alpha)$ is the gamma distribution (Forbes et al, 2011; Gelman et al, 2004). If $x \sim IG\ (\alpha, \beta)$, the mean and variance of the *IG* are expressed as follows.

$$E(x) = \frac{\beta}{\alpha-1}, \alpha > 1 \quad (6)$$

$$Var\ (x) = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}, \alpha > 2 \quad (7)$$

Normal. The normal or "Gaussian" family of distributions is typically used as a Bayesian analysis prior for coefficient parameters (i.e., intercepts and slopes) rather than variance components because it is symmetric and centered at its location parameter μ (mean) with a spread governed by its scale parameter σ (the standard deviation), and therefore can have any possible real value, including negative values (see Figure 5).

The well-known normal PDF of is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \quad (8)$$

for $-\infty < x < +\infty$ (Bandyopadhyay & Cherry, 2011; Kaplan, 2023).

Half-*t*. The half-*t* family of distributions (also known as the "folded-*t*") is also sometimes used as a prior distribution in Bayesian statistics. It is derived from the classic *t* distribution but with only the positive half represented (see Figure 6). Its location parameter is μ, its shape parameter is ν (degrees of freedom), and its width is dictated by the scale parameter σ, known as the standard deviation, which is positive. The scale parameter σ can be computed as $(\nu - 2)/\nu$ and its PDF can be expressed as:

$$f(x; \mu, \sigma) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi\sigma^2}\Gamma(\frac{\nu}{2})} \left(1 + \frac{(x-\mu)^2}{\nu\sigma^2}\right)^{-\frac{\nu+1}{2}}, \quad (9)$$

for $x \geq 0$, $\nu > 0$, $\sigma > 0$, $x \geq \mu$, where μ is the location parameter, σ is the scale parameter, and ν is the degrees of freedom (Hastie et al., 2009). When ν tends to ∞, the half-*t* becomes a half normal distribution. The mean and variance of the half-*t* are expressed respectively as follows:

$$E(x) = \mu + 2\sigma\sqrt{\frac{\nu}{\pi}}\frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})(\nu-1)} \text{ for } \nu > 1 \quad (10)$$

$$Var(x) = \begin{cases} \sigma^2\left(\frac{\nu}{\nu-2} - \frac{4\nu}{\pi(\nu-1)^2}\left(\frac{\Gamma\left(\frac{\nu+1}{2}\right)^2}{\Gamma\left(\frac{\nu}{2}\right)}\right)^2\right) & \nu > 2, \\ \text{Infinite} & 1 < \nu < 2, \\ \text{Undefined} & \nu \leq 1 \end{cases} \quad (11)$$

Half-Cauchy. Like the half-*t*, the half-Cauchy family of probability distributions is derived from the right half of the half-*t* with $\nu = 1$ (see Figure 7). As such, it is a 2-parameter distribution governed by location and scale parameters μ and σ, though $\mu = 0$ for most application. Given that is has the smallest possible ν, its tail is thick relative to other half-*t* distributions with $\nu > 1$.

The PDF of the half-Cauchy is given in Gelman et al. (2014) as:

$$f(x; \mu, \sigma) = \frac{2}{\pi\sigma}\frac{1}{1+(x-\mu)^2/\sigma^2} \quad (12)$$

for $x \geq 0$, $\sigma > 0, x \geq \mu$. Like half-*t*, the half-Cauchy is often used as a default prior for variance components in Bayesian MLM since it allows for a wide range of plausible values for the variance parameter like uninformative priors but can provide at least some information (i.e., as a weak informative prior) for the posterior estimation process (Gelman, 2006). Gelman (2006) as well as Polson and Scott (2012) noted that the half-Cauchy may be better than the inverse gamma (*IG*) as a variance prior for parameter values closer to zero.

Uniform. The uniform family of distributions, also referred to as the rectangular distribution, has parameters α or *a* (minimum) and β or *b* (maximum) and can be expressed as $X \sim U(\alpha, \beta)$. It is also known as a "flat" prior because every value within its boundary parameters has an equal probability of occurring (see Figure 8).

The PDF of the uniform distribution is given by Freund et al. (2010) as:

$$f(x) = \begin{cases} \frac{1}{\beta-\alpha} & \alpha \leq x \leq \beta, \\ 0 & x < \alpha \text{ or } x < \beta, \\ 1 & 0 \leq x \leq 1 \end{cases}, \quad (13)$$

with a mean of:

$$E(x) = \frac{\alpha+\beta}{2}, \quad (14)$$

and variance of:

$$Var(x) = \frac{(\alpha-\beta)^2}{12} \quad (15)$$

While the uniform is sometimes called an "uninformative" prior in the literature, it does contain information in its hyperparameters, so we term it as "weakly informative". Table 2 demonstrated the overview of the probability distributions and their priors discussed above.

Relationships among Distribution Families

While the previously described distributions are related, they are not necessarily equivalent. However, certain transformations and parameter conditions can lead to equivalences. Table 3 summarizes whether each distribution can be interchangeable or transformable.

For instance, the gamma distribution becomes the chi-square distribution when its shape parameter is an integer, and its scale parameter is 2. As well, the inverse gamma is a direct transformation of the gamma distribution. The following sections include an analysis of distribution transformability and non-transformability, aiding in the understanding of how these priors can be used in different software within the Bayesian analysis framework.

There are 12 transformable distribution family pairs I discussed: 1) the chi-square and gamma; 2) the chi-square and normal; 3) the chi-square and uniform; 4) the gamma and inverse gamma; 5) the gamma and normal; 6) the gamma and uniform; 7) the inverse gamma and normal; 8) the inverse gamma and uniform; 9) the normal and uniform, 10) the half-*t* and half-Cauchy, 11) the half-*t* and uniform, and 12) the half-Cauchy and uniform.

Pair 1: Transformability between Chi-square and Gamma

The chi-square is a special case of the gamma with shape parameter $\alpha = v/2$ (with v being the degrees of freedom of the chi-square), and rate parameter $\beta = 1/2$ (since the scale parameter, which is the inverse of the rate parameter, is 2) (Petrykowski, 2020). To show that gamma becomes the chi-square under certain conditions, consider the case when the shape parameter α of the gamma distribution is an integer and the rate parameter β is 2. The relationship between the two distributions under specific conditions can be explained through their PDFs (Mood et al., 1973).

1) *Set the Parameters*. To transition from the gamma to the chi-square, we set $\alpha = v/2$, and $\beta = 1/2$.
2) *Substitute Parameters into the gamma PDF*. Substituting these values into the gamma PDF as follows:

$$f(x; v/2\,, 1/2) = \frac{(1/2)^{v/2} x^{(v/2)-1} e^{-(1/2)x}}{\Gamma(v/2)}$$

$$= \frac{1}{2^{v/2}} \frac{x^{(v/2)-1} e^{-(1/2)x}}{\Gamma(v/2)}$$

$$= \frac{x^{v/2-1} e^{-x/2}}{2^{v/2}\Gamma(v/2)} \qquad (16)$$

3) *Recognize the chi-square PDF* (see Equation 17):

$$f_{Chi}(x; v) = \frac{x^{v/2-1} e^{-x/2}}{2^{v/2}\Gamma(v/2)} \text{ for } x > 0 \qquad (17)$$

Thus, the gamma with shape $v/2$ and rate $1/2$ is the chi-square with v degrees of freedom.

Pair 2: Transformability between Chi-square and Normal

If we have a chi-square random variable x with v degrees of freedom, the distribution of x is skewed to the right when v is small but becomes less skewed as v increases. To transform a chi-square variable to approximately follow a normal distribution, the transformation given by:

$$Z = \frac{(x-v)}{\sqrt{2v}} \quad (18)$$

where Z is the transformed variable that is approximately normally distributed with a mean of 0 and a variance of 1 as v approaches ∞ (commonly, $v > 30$ is used as a rule of thumb for approximation) (Penn State, n.d., "Standard Normal and Chi-Square"). This is because the mean $E(x) = v$, and the variance $Var(x) = 2v$. Thus, this formula allows us to standardize x to create a variable Z that has the properties of a standard normal distribution for large v.

Pair 3: Transformability between Chi-square and Uniform

To convert a variable from a chi-square to a uniform distribution, a series of steps that involved the properties of cumulative distribution functions (CDFs) can be followed (Ross, 2014a,b):

1) *Define the Chi-square distribution*. As discussed earlier, the PDF of a chi-square distribution variable X is given by:

$$f_X(x; v) = \frac{x^{v/2-1}e^{-x/2}}{2^{v/2}\Gamma(v/2)}, \quad \text{for } x > 0 \quad (19)$$

2) *Compute the CDF*. The CDF, $F_X(x)$ is the probability that the chi-square distributed variable is less than or equal to x. For the chi-square distribution, this function has a specific mathematical form that involves the gamma distribution due to the chi-square distribution with v degree of freedom is a special case of the gamma distribution, where the shape parameter α is $v/2$ and the scale parameter θ is 2 (Soch et al., 2024). Thus, the gamma function plays a central role in the mathematical expression of the CDF because the chi-square distribution is defined in terms of the sum of squares of v independent standard normal variables, leading to its connection with the gamma distribution. The CDF is given by the lower incomplete gamma function as follows:

$$F_X(x, v) = \frac{\gamma(k/2, x/2)}{\Gamma(k/2)} \quad (20)$$

where γ is the lower incomplete gamma, indicating the chi-square distributed variable is less than or equal to x.

3) *Apply the Probability Integral Transform*. If we have a continuous random variable X with CDF F_X, then the random variable $U = F_X(X; v)$ is uniformly distribution on $(0,1)$, where X is a chi-square distributed variable. Thus, the results are a set of U, which means each subinterval of equal length is equally likely to contain a U value within the interval $(0,1)$.

Pair 4: Transformability between Gamma and Inverse Gamma

The transformation of the gamma to the inverse gamma is given by taking the reciprocal of the variable.

1) *Change of Variables*. When we change variables from x to y, we also need to transform the PDF according to the change of variables formula in probability. For the transformation $y = 1/x$, the derivative dy/dx is $-(1/x^2)$, where dy/dx is the derivative of y with respect to x, and it represents the rate of change of y as x change. When changing variables, we take the absolute value of the derivative, so we get $|dy/dx| = 1/x^2$(Koch, 2007; Ma, 2017).

2) *Apply Change of Variables to Gamma PDF*. The transformed PDF using the variable y is expressed as follows (see Equations 21 and 22) (Cook, 2008; Niemi, 2021):

$$f_G(x;\ \alpha, \beta) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \quad \text{for } x > 0 \quad (21)$$

$$f_{IG}(y;\ \alpha, \beta) = f_G(x;\ \alpha, \beta) \left|\frac{dx}{dy}\right|$$

$$= \frac{\beta^{\alpha}\left(\frac{1}{y}\right)^{\alpha-1} e^{-\beta\left(\frac{1}{y}\right)}}{\Gamma(\alpha)} \frac{1}{\left(\frac{1}{y}\right)^{2}}$$

$$= \frac{\beta^{\alpha} y^{-\alpha+1} e^{-\frac{\beta}{y}} y^{2}}{\Gamma(\alpha)}$$

$$= \frac{\beta^{\alpha} e^{-\frac{\beta}{y}}}{y^{\alpha+1}\Gamma(\alpha)}$$

$$= \frac{\beta^{\alpha}}{\Gamma(\alpha)} y^{-\alpha-1} e^{-\frac{\beta}{y}} \quad (22)$$

3) *Recognize the inverse gamma PDF*. The PDF of the inverse gamma with parameters α and β is expressed as follows (see Equation 23):

$$f_{IG}(y;\ \alpha, \beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} y^{-\alpha-1} e^{-\frac{\beta}{y}} \quad \text{for } y > 0 \quad (23)$$

Thus, the inverse gamma PDF can be derived from the gamma PDF by the transformation $y = \frac{1}{x}$, which effectively inverts the variable.

Pair 5: Transformability between Gamma and Normal

Transforming a gamma to a normal distribution is not straightforward process, but we can approach this transformation through a limiting theorem, like the Central Limit Theorem (CLT). CTL provides a way to approximate a normal distribution from gamma when the shape parameter α becomes large (Ross, 2014a,b). The CLT states that for large sample sizes, the distribution of the sample mean approaches a normal distribution, regardless of the original distribution's shape, as long as the original distribution has a defined mean and variance (Chance et al., 2004; Chang et al., 2008). However, there's no strict rule for how large α should be, but the larger it is, the closer the approximation will be to a true normal distribution. The transformability depends on whether we are considering the rate parameter β or the scale parameter θ, each of which requires a different approach.

Approach 1 (considering the rate parameter β)

1) *Define Gamma Parameters.* For a gamma distributed variable X with a shape parameter α and a rate parameter β, the PDF is expressed as follows (see Equation 24):

$$f_G(x;\ \alpha, \beta)\ = \frac{\beta^{\alpha} x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \quad \text{for } x > 0 \tag{24}$$

2) *Calculate Mean and Variance of Gamma*.

$$\text{Mean } \mu = \frac{\alpha}{\beta} \tag{25}$$

$$\text{Variance } \sigma^2 = \frac{\alpha}{\beta^2} \tag{26}$$

3) *Apply CTL.* We standardize X to approximate a normal distribution from a gamma distribution when α is large. The process is as follows (see Equation 27):

$$Z = \frac{X-\mu}{\sigma} = \frac{X-\frac{\alpha}{\beta}}{\sqrt{\frac{\alpha}{\beta^2}}} = \frac{\beta X-\alpha}{\sqrt{\alpha}} \tag{27}$$

As α is large, Z will converge to a standard $N(0,1)$

Approach 2 (considering the scale parameter θ)

1) *Define Gamma Parameters.* For a gamma distributed variable X with a shape parameter α and a scale parameter θ, the PDF is expressed as follows (see Equation 28):

$$f_X(x; \alpha, \theta) = \frac{x^{\alpha-1} e^{-x/\theta}}{\Gamma(\alpha)\theta^{\alpha}} \quad \text{for } x > 0 \tag{28}$$

2) *Calculate Mean and Variance of Gamma*.

$$\text{Mean } \mu = \alpha\theta \tag{29}$$

$$\text{Variance } \sigma^2 = \alpha\theta^2 \tag{30}$$

3) *Apply CTL*. We standardize X to approximate a normal distribution from a gamma distribution when α is large, resulting in a variable Z that has a mean of 0 and a standard deviation of 1. The process is as follows (see Equation 31):

$$Z = \frac{x-\mu}{\sigma} = \frac{x-\alpha\theta}{\sqrt{\alpha}\theta} = \frac{x-\alpha\theta}{\sqrt{\alpha\theta^2}} \quad (31)$$

Pair 6: Transformability between Gamma and Uniform

Just like with the chi-square distribution, variables from a gamma can also be transformed into a uniform distribution using the CDF of the gamma distribution.

1) *Identify Variable*. As discussed above, we recognize that gamma distribution is defined by two parameters: a shape parameter, α and a scale parameter, β. Consider a variable X that follows a gamma distribution with known shape and scale parameters.
2) *Compute the CDF*. The CDF, $F_X(x)$ of the gamma distribution gives the probability that X will take a value less than or equal to x.
3) *Transform the Variable to a Uniform.* If X is a continuous random variable with CDF $F_X(x)$, then the random variable $U = F_X(x)$ is uniformly distributed on $(0,1)$. Apply this to a Gamma distributed random variable $X \sim \Gamma(\alpha, \beta)$, the transformed variable $U = G(X; \alpha, \beta)$ follows a Uniform distribution (Papoulis, 1965; Ross, 2014a,b).

Pair 7: Transformability between Inverse Gamma and Normal

The relationship between the inverse gamma and normal distribution can be transformed through an intermediary transformation on the square a normal random variable. The transformation process is as follows (Casella & Berger, 2002):

1) *Transform Inverse gamma to Gamma*. If X is an inverse gamma distributed variable with parameters α (shape) and β (scale), then the PDF of X is expressed as follows (see Equation 32):

$$f_X(x) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \text{ for } x > 0 \quad (32)$$

The transformation $Y = 1/X$ converts X to a gamma distributed variable Y with PDF:

$$f_Y(y) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{-\alpha-1} e^{-\frac{\beta}{y}} \text{ for } y > 0 \quad (33)$$

2) *Relate Normal to Gamma*. A normal variable $Z \sim N(0,1)$ can be transformed using Z^2, following a chi-square distribution with 1 degree of freedom (a special case of the gamma distribution):

$$f_{Z^2}(z) = \frac{1}{\sqrt{2\pi z}} e^{-z/2} \text{ for } z > 0 \quad (34)$$

This is equivalent to a gamma distribution with shape $\alpha = 1/2$ and rate $\beta = 1/2$.

To theoretically transform between the inverse gamma and normal distributions, the scaling and shifting parameters can be considered (Hogg et al., 2013). If we further transform Z^2 by scaling and shifting, $W = aZ^2 + b$ with appropriately chosen constants a and b, we can approximate a range of gamma distributions, though matching it directly to the specific gamma from the inverse gamma transformation Y (Penn State, n.d., "Sampling Distribution").

Pair 8: Transformability between Inverse Gamma and Uniform

Transforming an inverse gamma distributed variable to a uniform distribution still involves its CDF. This transformation is grounded in the probability integral transform, which states that the CDF of a continuous random variable, when applied to a sample from its own distribution, results in a uniformly distributed sample on the interval $(0,1)$ (Gelman, 2006; Levine, 2020).

1) *Define the Inverse Gamma*. As discussed above, the PDF of inverse gamma distributed variable X with parameters α (shape) and β (scale) is given by Equation 35:

$$f_X(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} e^{-\frac{\beta}{x}} \quad \text{for } x > 0 \tag{35}$$

2) *CDF of Inverse Gamma*. The CDF of the inverse gamma distribution is expressed in terms of the upper incomplete gamma function (see Equations 36 and 37):

$$F_X(x) = 1 - Q(\alpha, \beta/x) \tag{36}$$

$$Q(\alpha, \beta/x) = \frac{\Gamma(\alpha, \beta/x)}{\Gamma(\alpha)} \tag{37}$$

where $Q(\alpha, \beta/x)$ is the regularized gamma function (also called complemented incomplete gamma function), $\Gamma(\alpha, \beta/x)$ is the upper incomplete gamma function, and $\Gamma(\alpha)$ is the complete gamma function (MSU Libraries, n.d.; Wolfram MathWorld., n.d.). The function Q quantifies the probability that a gamma-distributed variable with shape parameter α is greater than β/x, providing the upper tail of the distribution.

3) *Probability Integral Transform*. By the probability integral transform, we get the Equation 38 as follows:

$$U = F_X(X) = 1 - Q(\alpha, \beta/X) \tag{38}$$

where the variable U follows a Uniform distribution on (0,1).

Pair 9: Transformability between Normal and Uniform

Variables from a normal distribution can be transformed into a uniform distribution using the CDF of the normal distribution. This process is based on the same principle used for transforming other types of distributions to a uniform distribution, known as the probability integral transform (Illowsky & Dean, 2018).

1) *Define the Normal Distribution*. The PDF for a normal distributed variable X with mean μ and standard deviation σ is expressed as follows (see Equation 39):

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{39}$$

2) *Identify a Variable*. Consider a variable X that follows the normal distribution with mean μ and standard deviation σ, denoted as $X \sim N(\mu, \sigma^2)$.

3) *Compute the CDF of the Normal Distribution*. The CDF, $F_X(x)$, for the normal distribution gives the probability that X will take a value less than or equal to x. The CDF of a normal distribution is given by:

$$F_X(x) = \frac{1}{2}\left[1 + erf\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right] \quad (40)$$

where erf is the error function.

4) *Apply the Probability Integral Transform*. If X is a continuous random variable with CDF $F_X(x)$, then the variable $U = F_X(X)$ will follow a uniform distribution on the interval [0,1] as follows (Hogg et al., 2013):

$$U = F_X(X) = \frac{1}{2}\left[1 + erf\left(\frac{X-\mu}{\sigma\sqrt{2}}\right)\right] \quad (41)$$

where the variable U follows a uniform distribution on (0,1).

Pair 10: Transformability between Half-t and Half-Cauchy (and vice versa)

The half-Cauchy can be seen as a special case of the half-*t* with degrees of freedom v = 1. To conceptualize the transformation from a half-*t* to a half-Cauchy distribution, consider the following steps (Balakrishnan & Nevzorov, 2003):

1) *Identify Parameters*. The transformation focuses on adjusting v in the half-*t*. Thus, suppose that we have a variable x with a half-*t* characterized by $v > 1$ and a scale parameter σ.

2) *Adjust v and Recognize the half-Cauchy as a Special Case*. We set a new distribution where $v = 1$ as recognizing the inherent relationship between these distributions based on their parameters.

Conversely, transforming a half-Cauchy distributed variable into a half-*t* distributed variable involves conceptualizing a transition to a distribution with more than one degree of freedom. Since the half-Cauchy is a half-*t* with v = 1, to "transform" it into another half-*t* distribution, we would define a new variable with the desired degrees of freedom greater than 1.

Pair 11: Transformability between Half-t and Uniform

The transformation from a half-*t* to a uniform distribution involves applying the CDF of the half-*t* distribution to the half-*t* distributed variable, converting it into a uniform distribution on $(0,1)$ (Psarakis & Panaretoes, 1990).

1) *Define the Half-t Distribution*. The half-*t* distribution is a special case of the *t*-distribution, restricted to positive values. It's typically derived from a *t*-distributed variable X where $|X|$ follows a folded *t*-distribution, and if $X \geq 0$, it specifically follows a half-*t* distribution (Ross, 2014a). The PDF of this distribution with degree of freedom v can be expressed as follows:

$$f_{half-t}(x, v) = \frac{2\Gamma(\frac{v+1}{2})}{\sqrt{v\pi}\Gamma(\frac{v}{2})}\left(1 + \frac{x^2}{v}\right)^{-\frac{v+1}{2}} \text{ for } x \geq 0 \qquad (42)$$

2) *Compute the CDF of the half-t Distribution*. The CDF, $F_{half-t}(x; v)$ for the half-*t* distribution gives the probability that X will take a value less than or equal to x. For the half-*t* distribution, this function is defined specifically to account for only the positive values of the *t*-distribution.
3) *Apply the Probability Integral Transform.* Apply the CDF of the half-*t* distribution to each value of X, obtaining $U = F_{half-t}(X; v)$. If X follows a half-*t* distribution, then U will be uniformly on $(0,1)$.

Pair 12: Transformability between Half-Cauchy and Uniform

Similarly, a half-Cauchy distributed variable can be transformed into a uniform distribution using the CDF of the half-Cauchy distribution (Balakrishnan & Nevrozov, 2003).

1) *Define the half-Cauchy Distribution*. Half-Cauchy distribution is a special case of the Cauchy distribution limited to positive values. it is a 2-parameter distribution governed by location and scale parameters μ and σ, though $\mu = 0$ for most application. The PDF of the half-Cauchy is given by:

$$f(x;\mu,\sigma) = \frac{2}{\pi\sigma}\frac{1}{1+(x-\mu)^2/\sigma^2} \quad (43)$$

2) *Compute the CDF of the half-Cauchy Distribution*. The CDF, $F_X(x)$, for the half-Cauchy distribution gives the probability that X will take a value less than or equal to x. For the half-Cauchy distribution, this function is defined from the CDF of the full Cauchy distribution, modified to account for only the positive values. The CDF is given by (Probability Distribution Explorer, n.d.):

$$f(x;\mu,\sigma) = \begin{cases} \frac{2}{\pi} arctan\left(\frac{x-\mu}{\sigma}\right) & x \geq \mu \\ 0 & otherwise \end{cases} \quad (44)$$

3) *Appy the Probability Integral Transform.* Apply the CDF of the half-*t* distribution to each value of X, obtaining $U = F(X;\mu,\sigma)$, where X is half-Cauchy distributed variable. The transformed variable U will be uniformly distributed on $(0,1)$.

The above discussed the direct conversion from one to the other distribution. The following sections discussed cases for non-transformable distributions.

Non-Transformable Distributions

Non-transformability between Chi-square and Inverse Gamma

As discussed above, the chi-square is a special case of the gamma with its shape parameter α set to $v/2$ and its scale parameter set to 2. The inverse gamma has a shape

parameter α and scale parameter β, which do not correspond to the chi-square's parameters. The inverse gamma is defined as the distribution of $1/x$ for a gamma-distributed variable x. This inversion does not result in a chi-square because it is not the inverse of the gamma (Wackerly et al., 2008).

Non-transformability between Chi-square and Half-t

The non-transformability between the chi-square and the half-*t* distributions due to their fundamentally different characteristics and the mathematical properties that define each distribution. The chi-square, characterized by its degrees of freedom and inherent right-skewness, becomes more symmetric with increasing degrees of freedom, lacking the heavy tails associated with the half-*t* (Wilson & Hilferty, 1983). In contrast, the half-*t*, derived from the absolute value of a t-distributed variable, features heavy tails and is defined for non-negative values. These intrinsic differences, including the distributions' skewness and tail behavior, underscore the lack of a mathematical pathway to transform variables between these distributions while preserving their unique characteristics.

Non-transformability between Chi-square and Half-Cauchy

These two distributions have no direct transformation due to the different support, tail behavior, and undefined moments. The half-Cauchy has extremely heavy tails such that its mean and variance are undefined (Probability Distribution Explorer (n.d.)). The chi-square, on the other hand, has a tail that decreases exponentially and has well-defined moments (e.g., its mean is equal to the degrees of freedom v, and its variance is $2v$) (Ramachandran & Tsokos, 2021). Thus, they are different in the mathematical properties and the statistical behaviors they model.

Non-transformability between Gamma or Inverse Gamma with Half-t or Half-Cauchy

The half-*t* has heavy tails for low degrees of freedom, which the gamma and inverse gamma distributions do not. In addition, the shape and rate parameters of the gamma and inverse gamma give them specific exponential decay behaviors, which do not match the polynomial decay of the half-*t*'s tails. Further, given that the half-Cauchy is a particular form of the half-*t*, it follows that the half-Cauchy cannot be transformed into the gamma or inverse gamma as well.

Non-direct transformability between Normal and Half-t

Transforming between a normal and a folded or half-*t* distribution is not a direct process due to their different properties and characteristics. The folded or half-*t* distribution is a modified t-distribution that only has the positive part of the distribution. The conceptual process can be expressed as follows (Psarakis & Panaretoes, 1990):

1) *Absolute Value*. Since the normal distribution is defined across the entire real line and the half-*t* is defined only for non-negative values, the first step involves taking the absolute value of the normal random variable, $Y = |x|$. This step will mirror the negative values to the positive side, doubling the density for positive values.
2) *Scaling Transformation*. Next, the absolute value transformed variable could be scaled to achieve the heavier tail behavior characteristic of the half-*t* distribution. The scaling factor would vary with the value to reflect the heavier tails of the half-*t* distribution.
3) *Degrees of Freedom Adjustment*. The half-*t* has a shape parameter - degrees of freedom (v) - that affects the tail heaviness. Unlike the normal distribution, whose tails decrease exponentially, the *t*-distribution and its half-*t* variant have tails that follow a polynomial decay. Adjusting for the distribution of Y to more closely resemble the decay rate of the half-*t* distribution with the desired v is required.

4) *Simulation.* In practical, rather than transforming a normal variable into a half-*t* variable, it is more efficient to simulate data directly from a half-*t* distribution. This involves sampling from a *t*-distribution with the desired ν, taking the absolute value of the samples to obtain a folded *t*, and then using this distribution as a half-*t*.

Non-direct transformability between Normal and Half-Cauchy

Similarly, transforming a normal into a half-Cauchy distribution is not a straightforward process due to the fundamentally different properties of these two distributions. The normal distribution is symmetric with light tails, while the half-Cauchy is a one-sided distribution with heavy tails (Polson & Scott, 2012). To simulate a transformation from a normal variable to a variable that mimics the behavior of a half-Cauchy, the following steps could be taken:

1) *Absolute Value*. Since the half-Cauchy is one-sided (defined only for non-negative values), begin by taking the absolute value of the normally distributed random variable x, This gives a new variable $Y = |x|$, which is the folded normal distribution that only takes non-negative values, similar to the half-Cauchy in that respect.
2) *Heavy Tails*. The folded normal distribution does not exhibit the heavy-tailed nature of the half-Cauchy. To imitate the heavy-tailed behavior, we have to adjust the PDF. A half-Cauchy distribution has a PDF that decreases according to a power-law as we move away from the location parameter, as opposed to the exponential decrease of a normal distribution's tails.
3) *Scaling Transformation*. While the absolute value folds the negative side of the normal distribution onto the positive, creating a heavy tail from a normally distributed variable is not something that can be done with a simple transformation because the normal distribution decays too quickly. A scale transformation that could, in theory,

mimic the heavy tails of a half-Cauchy would need to be non-linear and dependent on the value of the variable itself, which is not straightforward.

CHAPTER FOUR:

SOFTWARE FOR BAYESIAN ANALYSIS

In the evolving field of Bayesian analysis, software tools cater to different needs and preferences (Mai & Zhang, 2018). Table 4 summarizes these different software in alphabetical order; below I briefly describe each in turn.

ADMB or **AD Model Builder** is a free and open-source software for non-linear statistical modeling (Monnahan et al., 2014; Monnahan & Kristensen, 2018). The "AD" stands for "automatic differentiation," enabled by the AUTODIF Library, a C++ extension by Fournier, implementing reverse mode automatic differentiation. ADMB integrates MCMC methods and supports modeling random effects in a frequentist framework using Laplace approximation and importance sampling (Monnahan et al., 2014). It is widely used in in ecological and resource management contexts. ADMB is available for Windows, Linux, Mac, and OpenSolaris. The R2admb package allows users to interface ADMB within *R*.

BayesiaLab (Conrady & Jouffe, 2015) is a software package designed for manipulating and visualizing Bayesian network analysis, which is available for Windows, Mac, and Unix. It provides a comprehensive environment for machine learning, knowledge modeling, analytics, simulation, and optimization, all based on the Bayesian network paradigm.

The **BUGS** project (which stands for "Bayesian inference Using Gibbs Sampling") was initiated in 1989 by the MRC Biostatistics Unit at the University of Cambridge. It aimed to create flexible, user-friendly software for Bayesian analysis of complex statistical models using MCMC algorithms - Gibbs Sampler for efficient computation. The most well-known version, WinBUGS, is a point-and-click software designed for operating machines using Windows (Lunn et al., 2000). OpenBUGS offers a similar functionality and can operate on machines with

Windows, Mac OS, or Linux (Lunn et al., 2009). MultiBUGS is a software extension designed to run on parallel computing environments, enhancing the efficiency of Bayesian inference by splitting the workload across multiple processors (Goudie et al., 2020). This tool is particularly useful for handling more computationally intensive models.

For scenarios requiring speed over traditional Bayesian MCMC estimation methods, **INLA** (Rue et al., 2009) provides integrated nested Laplace approximations, ideal for handling latent Gaussian and hierarchical models, and featuring integration with *R* as R-INLA package (https://www.r-inla.org/).

JAGS (which stands for "Just Another Gibbs Sampler") (Plummer, 2003) is designed for the general analysis of Bayesian graphical models. JAGS is written in C++ and its language is similar to the BUGS family of programs. JAGS uses a combination of Metropolis sampling, Gibbs sampling, and other MCMC algorithms. Rjags is an *R* package that provides an interface to JAGS, enabling users to run JAGS from within the *R* environment. JAGS is widely used in statistical research and applications due to its ability to handle a wide variety of models and data structures efficiently.

Mplus (Muthén & Muthén, 1998-2017) is a comprehensive statistical modeling program that supports a wide range of analyses, including Bayesian analysis using MCMC Gibbs sampling. The algorithm in Mplus is based on use of conjugate priors (Asparouhov & Muthén, 2010a,b). Key features include flexibility in handling complex models (e.g., SEMs, MLMs), specification of prior distributions, advanced MCMC estimation, posterior predictive checking, and diagnostic tools for evaluating model fit and convergence.

PyMC is a Python library designed for Bayesian statistical modeling and probabilistic machine learning, where users can specify prior distributions, likelihood functions, and compute

posterior distributions using advanced MCMC sampling algorithms. PyMC offers a user-friendly interface for defining models using Python's syntax and integrates with other Python packages (e.g., NumPy, SciPy, and pandas), making it to be a powerful tool for Bayesian analysis in research and applied settings (Salvatier et al., 2016). In addition, PyMC also includes visualization tools, like ArviZ that could provide extensive plotting capabilities for posterior distributions, trace plots, and diagnostic checks (Kumar et al., 2019; Martin, 2018).

R (R Core Team, 2023) is a widely used programming language and environment specifically designed for statistical computing and graphics. For Bayesian analysis, R provides extensive support through various packages, including *brms* (Bürkner, 2017a,b), *coda* (Plummer et al., 2006), *rstan* (Guo et al., 2020), and *rstanarm* (Goodrich et al., 2020). The next chapter will provide more details for the descriptions and features of each of these packages.

SAS/STAT (SAS Institute Inc., 2020) incorporates Bayesian procedures within the SAS software suite through its PROC MCMC procedure, which supports the definition and estimation of Bayesian models using MCMC methods like Gibbs Sampler and Metropolis-Hastings methods (Stokes et al, 2014). Within SAS/STAT, users can specify complex hierarchical models, choose from a variety of prior distributions, and perform posterior predictive checking. SAS/STAT also provides diagnostic tools to assess the convergence and accuracy of MCMC simulations, including trace plots, autocorrelation plots, Gelman-Rubin statistics, Geweke diagnostics, effective sample size, posterior summaries, and density plots.

Stan is an open-source C++ program that has a user interface similar to BUGS and employs advanced MCMC methods like Hamiltonian Monte Carlo (HMC) (Chen et al., 2014) and its variant, the No-U-Turn Sampler (NUTS) for more efficient computation (Carpenter et al., 2017). HMC is a generalization of the Metropolis–Hastings algorithm, and it adapts based on the

sampler's position and the gradient of the posterior distribution (Betancourt, 2019). NUTS extends HMC by optimizing the sampling process, allowing for efficient exploration of the parameter space without the inefficiencies that can occur with standard HMC (Hoffman & Gelman, 2014). A detailed introduction to HMC and NUTS is beyond the scope of this dissertation. For more detailed information on HMC and NUTS, refer to Betancourt (2019), Betancourt and Girolami (2015), and Hoffman and Gelman (2014). Importantly, Stan provides a user-friendly interface with *R*, and many *R* packages now use Stan such as *brms*, and includes comprehensive diagnostic tools to assess convergence and accuracy of the sampling process.

Last, **Turing** (Ge et al., 2018) was built on the Julia programming language designed for Bayesian statistical modeling and inference. It implements many MCMC methods, including HMC, NUTS, and particle Gibbs sampling. Turing's integration with Julia allows it to use Julia's high-performance capabilities for computationally intensive tasks. The language is designed to be intuitive for users familiar with Julia, and it supports the direct specification of probabilistic models using Julia's syntax.

CHAPTER FIVE:

DEFAULT VARIANCE PRIORS IN BAYESIAN MLM SOFTWARE

Bayesian multilevel modeling (MLM) software utilize different default priors, likely due to their unique discipline-specific histories. Table 5 summarizes the *variance prior* defaults of the major statistical packages used for Bayesian MLMs, with a focus on *R* packages given that these are open-source, along with other currently popular packages. Below I provide some additional details.

R Packages

In *R*, the **brms** package (Bürkner, 2017a,b) is based on the open-source software *Stan* (Carpenter et al., 2017; named after the famous mathematician Stanisław Ulam) and uses the half-*t* with v = 3 degrees of freedom and scale parameter based on the observed data estimated standard deviation (minimum of 2.5) for default scalar variance priors. For variance-covariance matrices such as MLMs with random intercepts and slopes, its default is the LKJ multivariate distribution with a hyperparameter based on the Cholesky decomposition of the information matrix, also known as Jeffrey's prior.

The **BRugs** package in *R* (Thomas et al., 2006) implements one of the first open-source Bayesian software available, known as *Bayesian Analysis Using the Gibbs Sampler (BUGS*) which incidentally has also gone by *WinBUGS* and *OpenBUGs*. Although it does not have a default, the available scalar variance prior distributions are gamma and inverse gamma, with some tutorials suggesting use of inverse gamma with hyperparameters a and b set at 0.001, 0.001 (e.g., Ntzoufras, 2011)

Like the brms package, the *R* **rstanarm** package (Gabry & Goodrich, 2020) is based on *Stan* and uses the LKJ multivariate distribution as the default variance-covariance prior. The *R*

runjags package (Denwood, 2016), based on *Just Another Gibbs Sampler* (*JAGS*; Plummer, 2003), allows users to select scalar variance priors based on the DuMouchel or half-Cauchy distributions (in other words, it does not have a default).

Other Popular Software

In *MLwiN* stand-alone software with Bayesian MCMC estimation (Browne & Rasbash, 2009), the scalar variance prior default is the gamma prior distribution with very small values of a and b to mimic a uniform distribution. For variance-covariance matrices, the prior default is the Wishart multivariate distribution that uses some form of a likelihood-based estimate of the precision matrix (inverse of information matrix) for hyperparameters, which makes it weakly informative.

Similar to *MLwiN*, *Mplus* stand-alone software (Muthén & Muthén, 2017) uses the inverse gamma with a and b hyperparameters set to –1, 0 as the default prior for scalar variance (which is essentially a Uniform prior with a and b set to 0, ∞). For variance-covariance matrices, the inverse Wishart distribution with (0, –*p*–1) is the default, where *p* is the dimension of the multivariate block of latent variables.

Finally, like the brms and rstanarm *R* packages, the stand-alone *SAS*/*STAT* software uses the LKJ distribution using Jeffrey's prior as its default prior for Bayesian MLM variance-covariance components (SAS Institute Inc., 2020; Tiao & Box, 1973).

In summary, across the current Bayesian MLM software available, default priors for scalar variance component priors include: gamma, inverse gamma, half-*t*, and half-Cauchy. For multivariate variance-covariance priors (i.e., MLMs with intercept and slope variances *and covariances* estimated), weakly informative LKJ and Wishart distributions are used. For this dissertation, I focused on scalar variance priors.

CHAPTER SIX:

RESEARCH ON BAYESIAN MLM VARIANCE PRIORS

Many research studies have now shown the issues involved in modeling multilevel data using frequentist estimation in circumstances where there are few clusters (i.e., a small level 2 sample size); in these circumstances a Bayesian analysis may be more advantageous (see, e.g., Baldwin & Fellingham, 2013; Browne & Draper, 2006; Gelman, 2006; Lee et al., 2004; McElreath, 2020; McNeish, 2016a; Stegmueller, 2013). This said, the prior choice for a variance component for small samples requires more careful consideration, as the prior distribution will have a greater impact on the posterior distribution than for large samples. Table 6 summarizes the major findings from previous research on Bayesian variance priors.

Although an *uninformative* uniform prior with large variance or an inverse gamma prior with small hyperparameters may be software defaults and have been recommended by others (e.g., van de Schoot et al., 2015) as they are both considered "flat" distributions, these priors still actually contain information and there is no guarantee the algorithm will not select extreme values from them and distort variance estimates (either with values that are too high, such as when using the uniform distribution, or too low, such as when using the inverse gamma with tiny hyperparameters).

While several studies have focused on comparing frequentist and Bayesian analysis results (e.g., Browne & Draper, 2000, 2006; Farrell & Ludwig, 2008; McNeish, 2016a; Stegmueller, 2013), few have actually compared the performance of different priors for variance components. First, Gelman (2006) found that flat uniform priors typically performed well for $J \geq 3$ clusters, which would include most cases of MLMs, but also recommended use of the half-*t* or half-Cauchy over the uniform to avoid extreme values. Later, Baldwin and Fellingham (2013)

found that using a Gamma informative prior performed better than a uniform prior. McNeish and Stapleton (2016a) found that using an inverse gamma prior with small hyperparameters (*IG*(0.01,0.01)) or a half-Cauchy prior with modest variance (half-Cauchy(0,4)) minimized variance component estimate bias, but they also noted that the inverse gamma prior tended to underestimate variance whereas the half-Cauchy tended to overestimate it. Research by Bolin et al. (2019) concurred with McNeish and Stapleton (2016a) and suggested that inverse gamma priors were best at estimating variance components. Daniels (1999) suggested employing a uniform shrinkage prior used for variance components in MLMs by placing a uniform distribution to the shrinkage parameter $S = \sigma^2/(\sigma^2 + \tau^2)$, where σ^2 represents the within-group variance and τ^2 the between-group variance. This approach results in a prior for τ^2 formulated as $\sigma^2/(\sigma^2 + \tau^2)^2$. Finally, Zheng et al. (2023) found that informative variance priors could yield biased results, but the flaw in their report is that they used normal priors for variance components, which is generally considered inappropriate for variance component estimation.

In short, the evidence to date has been mixed and could be considered confusing to applied analysts. Further, the reliance on software defaults or rules-of-thumb for prior distributions can be problematic (McNeish, 2016a). Thus, three key challenges remain for selecting *informative* variance priors:

1) there is a disproportionate influence of priors on posterior distributions in smaller samples, which can occur relatively often in educational research, and the impact of incorrect priors can result in highly inaccurate parameter estimates (Gelman, 2006; McNeish, 2016a,b; Smid et al., 2020a,b; Smid & Winter, 2020);
2) there are differences among software packages for prior options and defaults, which may be confusing for applied researchers who are not familiar with all of the different

distributions or their relationships, leading to incorrectly assuming that the default prior is the best way to have an objective analysis (Holtmann et al., 2016; Smid & Winter, 2020; van Erp & Browne, 2021; van Erp et al., 2018);

3) and finally, a common misunderstanding is that default priors are "uninformative", yet most priors do carry implicit information that can affect the posterior distribution and, consequently, inferences about the meaning of results (Betancourt, 2017; McNeish, 2016a,b). This issue is further complicated by the variability of default priors across different statistical software, leading to disparate results and potentially introducing bias based on software choice.

In short, researchers may believe they are analyzing data in a manner that allows "the data to speak" when essentially, they are letting "the default priors to speak." In the following chapter I summarize some current practices in applied Bayesian MLM research.

CHAPTER SEVEN:

REVIEW OF APPLIED RESEARCH USING BAYESIAN ANALYSIS

Applied research, particularly in educational and social sciences, often utilizes a variety of software tools for Bayesian analysis. Prominent among these are JAGS, MLwiN, Mplus, R (with packages like brms, rstanarm, and runjags), Stan, and WinBUGS. Recent literature reviews highlight the widespread use of these tools (e.g., Araya et al., 2018; Bressler et al., 2021; Hoogerheide et al., 2012); Jephcote et al., 2021; Kolovou et al., 2021; Lipscomb et al., 2019; Nalborczyk et al., 2019; Scarpino et al., 2011).

The primary Bayesian model estimation method employed across is MCMC, a set of algorithms used to sample from a probability distribution. Studies often use software environments like JAGS and WinBugs, which commonly implement Gibbs sampling – a type of MCMC algorithm particularly suited for models with conditional distribution that are easy to sample from (Notenboom & Reitsma, 2007). Gibbs sampling is advantageous when the full conditional distributions of the parameters are known and can be sampled from directly. For example, Gelfand et al. (1990) utilized Gibbs sampling in WinBUGS to estimate a Bayesian hierarchical model for spatial data, where the inverse gamma prior was used for variance components. Similarly, Kéry and Schaub (2011) applied Gibbs sampling in WinBUGS for a Bayesian analysis of population dynamics, using hierarchical models to estimate survival probabilities and reproduction rates with normally distributed priors on fixed effects and inverse gamma priors on random effects variances. This approach allowed for the accommodation of both individual-level and population-level uncertainties. Further, Spiegelhalter et al. (2002) demonstrated the use of Gibbs sampling in Bayesian model determination using WinBUGS to

health data where model parameters were given vague normal priors, and variance components were assigned inverse gamma priors.

The use of Gibbs sampling in these software environments is widespread due to its simplicity and efficiency in handling a range of Bayesian hierarchical models. The choice of priors, such as normal distributions for coefficients and inverse gamma distributions for variances, often reflects a balance between easy of calculation and the incorporation of prior knowledge. I conducted an informal search of applied research in ERIC (Educational Resources Information Center) and Google scholar using keywords "Bayesian" and "multilevel model" and limiting the search from January 1, 2001 to December 31, 2023. Of the list that appeared, I reviewed the top 34 articles that appeared on the list and were truly applications of Bayesian analysis of multilevel data (i.e., not methodological research).

Among these articles, I found that the use and reporting of priors varied widely. For instance, Ayaya (2018), Kolovou et al. (2021), Lipscomb et al. (2019), and Rodríguez (2019) employed Bayesian MCMC estimation of their MLMs but did not provide details on the priors used. Others reported using uninformative or software default priors, including Jin and Rubin (2009), Doyle (2010), Doyle and Gorbunov (2011), Galbraith et al. (2012), Galbraith and Merrill (2012), Gudmestad and Geeslin (2013), Tamminen (2016), Tafvelin (2021), and Wheadon (2013). There were, however, some studies that adopted weakly informative priors to guard against overfitting and improve model performance. For example, Nalborczyk (2019) specified weakly informative priors for different models using brms in *R*. Particularly, both parameters – α (intercept) and β (slope for gender) are given a vague (weakly informative) Gaussian prior centered at 0 with a standard deviation of 10. For the residual variation, a half-Cauchy(0,10) was used. Skinner et al. (2022) used weakly informative normal priors in *Stan* for intercept and slope

$(\beta_0, \beta_{female} \sim N(0,2))$ and random effects $(\alpha \sim N(0,\sigma))$; and a truncated standard normal prior $N_+(0,1)$ was set for variance term for the group random effects. Similarly, Skinner and Doyle (2022) assigned weakly informative normal priors $N(0,\sigma)$, where σ is the standard deviation of the priors for the regression coefficients (α, β, γ), and the standard deviation of these parameters is given a truncated normal prior $N_+(0,1)$.

Buckley and Schneider (2005) used weakly informative priors in *WinBUGS*. Specifically, each π_{ks}, the probability of success for each binomially distributed subgroup, is given a weakly informative normal prior $N(0,\sigma)$; the standard deviation σ used in the priors for π_{ks} is itself modeled with a truncated normal distribution $N_+(0,1)$. Last, Bressler (2021) set moderately informative priors using rstanarm in *R*. Specifically, they employed the Bayesian multilevel logistic regression model using a series of priors, including intercept and condition-specific intercept adjustments are assigned a moderately informative prior $N(0,10)$; cluster-specific intercept adjustments also follow a normal distribution, the standard deviation $(\sigma_{cluster})$, however, is modeled separately: $N(0,\sigma_{cluster})$. The flow effect coefficient is given a moderately informative normal prior $N(0,2.5)$, and the variance parameters for the cluster and condition random effects are modeled using exponential distributions with a scale of 1.

In short, many researchers appear to rely on default priors (as already discussed in McNeish, 2016a) or simply do not report their Bayesian analysis technical details as they should. Using simulation and an applied demonstration in the next chapters, I aim to provide analysts with guidance for use of informative variance priors.

CHAPTER EIGHT:

MONTE CARLO SIMULATION STUDY

Data Generation

Conditions

For data generation and analysis, I used *Mplus 8* (Muthén & Muthén, 2017) and *MplusAutomation* (Hallquist & Wiley, 2018) in *R*. I focused on 2-level models with two predictors (*X*s) and one continuous outcome *Y* assumed to be measured at L1. Intraclass correlations (ICCs) – the level of correlation among scores within clusters – were generated using pre-specified L1 and L2 variance components (see next section). The manipulated factors were as follows.

1) three levels of the number of clusters: $J = 10$, 30, and 100 clusters to simulate real world small- and large-scale data conditions, such as a small samples of students in longitudinal studies (where L1 is measurements) or small samples of schools in cross-sectional studies (where L1 is students or teachers). These levels represent different scales of educational settings, ranging from small groups to large cohorts typical of national assessments (Raudenbush & Bryk, 2002). Cluster sizes of $J = 10$ mimics small-scale studies often seen in pilot projects or in studies with limited populations, such as special education programs. Cluster sizes of 30 align with medium-sized studies, common in regional educational assessments or evaluations of new curricula across several schools. 100 clusters are representative of larger-scale research, typical in national educational assessments or comprehensive program evaluations across multiple regions or states.

2) two cluster sizes: M = 5, 30 to simulate common educational groups in real world conditions, such as learning in small groups, or classrooms, respectively, which differ in their internal dynamics and the resulting intraclass correlations (Hedges & Hedberg, 2007);
3) three intraclass correlation sizes ICCs: .01, .20, and .40 to represent null, typical, and large sized dependencies in L1 units within L2 units in education settings (e.g., Hedges & Hedberg, 2007) (without loss of generality but for simplicity, with X and Y ICCs set to be the same);
4) three L1 total R^2 levels: .0, .20, and .50 to represent null, small, and large sized effects (e.g., McNeish et al., 2016a; Stapleton, 2007) (without loss of generality but again for simplicity, Xs were assumed to be uncorrelated with each other); and
5) three L2 total R^2 levels: .0, .20, and .50 to represent null, small, and large sized effects (e.g., Stapleton et al., 2016a,b) (again, Xs assumed uncorrelated).

These five factors were fully crossed, for a total of 162 conditions. For each condition, I simulated 200 datasets (1,000 is standard but due to time constraints I used 200, as Bayesian analyses take more time than frequentist analyses).

2-Level Latent Decomposition Model

ICCs were induced by setting specific values at L2 for the variance of L1, each calculated a-priori for every combination of L1 and L2 $R^2_{y.x}$, with initial coefficient values derived using eigenvalue decomposition. Specifically, we assumed predictors have separable within- and between-cluster effects on the outcome using latent disaggregation, as follows:

$$Y_{ij} = \gamma_{00} + \gamma_{10}\left(X_{1ij} - \mu_{1.j}\right) + \gamma_{01}\left(\mu_{1.j}\right) + \gamma_{20}\left(X_{2ij} - \mu_{2.j}\right) + \gamma_{02}\left(\mu_{2.j}\right) + u_{0j} + \varepsilon_{ij}\ , \quad (45)$$

where: Y_{ij} is the normal outcome for i^{th} score in the j^{th} cluster; γ_{00} is the grand mean, or the overall average effect across all clusters when all predictors are at their mean values; γ_{10} and γ_{20}: the within-cluster effects (slopes) of the $X_1$and X_2 predictors, respectively. These effects are calculated after centering the predictors within each cluster ($X_{1ij} - \mu_{1.j}$ and $X_{2ij} - \mu_{2.j}$), which means subtracting the cluster mean from each individual score within the cluster; γ_{01} and γ_{02}: the between-cluster effects (slopes) of the $X_1$and X_2 predictors, respectively. These effects use the cluster means of the predictors, $\mu_{1.j}$ and $\mu_{2.j}$; u_{0j} is the deviation between the cluster mean and the grand mean of the intercept; ε_{ij} is the residual error for the i^{th} score in its j^{th} cluster.

Methods

Parameter Estimation Approaches

For each of the 200 replicates per condition, data were analyzed with a correctly specified model using seven parameter estimation approaches, as follows:

1) frequentist full information maximum likelihood;
2) Bayesian estimation with flat (largely **uninformative**) priors, which is the default in *Mplus* and *blavaan*, for both the fixed effects (slope coefficients) and random effects (variance components), with a normal distribution *N*(0, 10,000) for the slope coefficients and inverse gamma *IG*(-1, 0) for the variance components;
3) Bayesian estimation with **weak informative fixed effect priors** for the L1 and L2 slope coefficients using *N*(true value, 10,000);
4) Bayesian estimation with **weak informative random effect priors** for the L1 and L2 residual variance terms using *IG*(.01,.01) as recommended in previous literature;

5) Bayesian estimation with **strong informative random effect priors** for the L1 and L2 residual variance terms using *IG*(a,b), where a and b are derived from setting the distribution mode (peak) to the true variance values;
6) Combining the weak fixed effect with weak random effect priors (#3 and #4); and finally,
7) Combining the weak fixed effect with strong random effect priors (#3 and #5).

The important contribution here is that there is no literature to date that has compared use of strong informative *IG* variance priors. I derived these priors beginning with a scaled (weighted) chi-square distribution (theoretically known sampling distribution for variances), and then translating the chi-square distribution to Gamma (*G*) and then *IG*.

Results

In all results that follow, I report median posterior estimates with equal-tailed 95% credible intervals.

Level 1 Parameter Estimates

As can be seen in Table 7, there were no meaningful differences among the model estimation approaches in terms of bias or false positive the L1 predictor slope coefficients or residual error. This is not surprising, since all the replicates had modest or large L1 sample sizes.

Level 2 Parameter Estimates

For each estimation approach and L2 predictor effect size (collapsed across L1 predictor effect sizes), false positive rates and raw bias for the L2 intercept coefficient are shown in Table 8 and Figure 9 and Figure 10, respectively. Bias for the L2 intercept variance is illustrated in Figure 11. As can be seen, false positive error was well-maintained across all estimation approaches (within the nominal value of 5% ± 5% (Hoogland & Boomsma, 1998), and correct

positive was consistently better for Bayesian-estimated models compared to frequentist maximum likelihood, particularly for small samples (i.e., over 80%). False positive/correct positive and raw bias for the L2 slope coefficient are shown in Table 9 and Figure 12 and Figure 13, respectively. The same pattern observed for the intercept was also found for the slope.

Perhaps the most striking finding is that the weak variance prior approach, *IG*(.01,.01), produced nearly identical results to those of the more strongly informative priors, *IG*(a,b) in terms of false positive, correct positive, and bias. This may be because the less informative prior is in fact still informative, with a mode near 1 and a very high peak at the mode (resulting in a tight variance) whereas the strongly informative prior, whose mode was near the true variance, has a softer peak and therefore greater variability in its distribution. A second potential reason for the lack of difference is that the simulations used unit-normal variables which have variances close to 1 or less. In the next paragraph, I illustrate the methods using larger scaled data.

To explore this idea, the same data generation and analysis procedures were used with *X* and *Y* variables scaled to have variances equal to 10,000 (i.e., *SD* = 100), similar to large-scale international tests. However, for this exploratory analysis only 20 replications were specified. Across all conditions and irrespective of whether informative coefficient priors were specified, results showed that the true L2 variance estimate was underestimated by an average of 19% for *IG*(.01,.01) but 0% for *IG*(a,b), but the disparity was particularly pronounced for smaller sized L2 samples (see Figure 14).

Last but not least, because all the simulation results thus far have assumed the population values were known for setting up the informative priors, I re-examined population value coverage for the *IG*(a,b) informative variance priors using sample-based priors instead. Again, given the length of time that these analyses require, I focused on datasets for the *J* = 30 set of

conditions and re-analyzed each dataset with uninformative priors first, and then in a second step, used the *sample-based variance estimate results* from the first step as informative variance priors (no informative prior were used for coefficients); the brms package was used for these analyses, with equal-tailed intervals reported. Results, shown in Table 10, indicated slight undercoverage (92-93%) using the sample-based priors compared to the uninformative prior analyses (which averaged 95-96% coverage). The trade-off, of course, is in the width of the intervals for the L2 variance, which was on average 17% lower for the informative variance prior using inverse gamma compared to an uninformative variance prior (and more for the smaller cluster size condition of $M = 5$ compared to $M = 30$).

CHAPTER NINE:

REAL DATA APPLICATION EXAMPLE

Research Context and Dataset Description

To demonstrate use of different priors with authentic educational data, I used the dataset from the *Trends in International Mathematics and Science Study* (TIMSS), an assessment administered by the International Association for the Evaluation of Educational Achievement. The assessment focuses on the competencies of fourth- and eighth-grade students in mathematics and science across countries (Mullis et al., 2020). These data are naturally multilevel in structure, with items/subscales nested within students, nested within schools, and within countries. Because my focus is on smaller sized L2 samples, I focused on the U.S. only and further took a subsample of schools and students to compare Bayesian informativeness levels similar to the simulation study. For the demonstration, I focused on examining how Grade 8 math achievement predicts Grade 8 science achievement. This focus is grounded in existing literature that has consistently demonstrated significant correlations between performance in mathematics and science subjects, reflecting underlying shared competencies and the reinforcing nature of these disciplines on cognitive development (e.g., Claessens & Engel, 2013; Singh et al., 2002). Last, I would like to note that the TIMSS 2019 data were used specifically because it was the last wave that occurred before the pandemic, providing a better baseline assessment of achievement.

The TIMSS 2019 International Database offers five distinct estimates of individual student scores for each achievement scale, known as “plausible values.” These values are not singular test scores but a set of statistically derived estimates that reflect the uncertainty in scale estimation, thereby capturing the range of possible achievement levels for each student (Fishbein et al., 2021). The plausible values represent the most reliable metric for student achievement on

the respective scales within the TIMSS 2019 International Database and are thus recommended as the primary outcome measure for analyses exploring student performance.

The dataset I used includes a total of 8,698 students (L1 units) nested within 273 schools (L2 units). For simplicity, I used the first plausible value for math and science scores. The variables were as follows:

- BSSSCI01: Science achievement score (outcome variable), the first plausible value in the publicly available dataset;
- ZBSMMAT01_CMC: Cluster-mean centered and z-scored student mathematics score (first plausible value), serving as the L1 predictor;
- ZBSMMAT01_agg: Cluster-aggregate and z-scored mathematics score, serving as the L2 school-wide academic environment context predictor; and
- Interaction: Between L1 and L2 predictors.

Again, because the full dataset is large, I used a random subsample to better assess the Bayesian estimation prior choices for much smaller sized samples (as this is the focus of my work). Specifically, I used the dplyr package in *R* with a random seed 2949 to randomly select 30 schools, and from each selected school, five students. For the subsample, I also re-centered and z-scored predictors.

Methods

Model Specification

I used a 2-level random intercept model for each analysis, as follows.

$$\begin{aligned}\mathrm{BSSSCI01}_{ij} = \gamma_{0j} + \gamma_{1j}\left(\mathrm{ZBSMMAT01_CMC}_{ij}\right) + \gamma_{2j}\left(\mathrm{ZBSMMAT01_agg}_{j}\right) \\ + \gamma_{3j}\left(\mathrm{ZBSMMAT01_CMC}_{ij} \times \mathrm{ZBSMMAT01_agg}_{j}\right) + u_{0j} + \varepsilon_{ij} \qquad (46)\end{aligned}$$

In the model above, $\mathrm{BSSSCI01}_{ij}$ is the science achievement score for student i in school j; $\mathrm{ZBSMMAT01_CMC}_{ij}$ is the cluster-mean centered and standardized mathematics score for student i in school j; $\mathrm{ZBSMMAT01_agg}_{j}$ is the aggregate mathematics score for school j, calculated as the mean of the mathematics scores of all students within the school, z-scored at the school level; γ_{0j} is the intercept for school j, indicating the average science score in school j when all predictors are at their mean values; γ_{1j}, γ_{2j}, γ_{3j} are coefficients for student-level mathematics scores, school-level aggregate mathematics scores, and their interaction, respectively, varying at the school level; u_{0j} is the school residual error, and ε_{ij} is the within-school (student level) residual error. Both residuals are assumed to be normally and independently distributed.

Bayesian Estimation

I modeled the data using the same prior approaches as used in the previously described Monte Carlo simulation. Specifically, data were analyzed with a correctly specified model using six Bayesian parameter estimation approaches via the *R* brms package. Importantly, I started with an uninformative uniform prior to extract the initial parameter estimates that were then used for the informative variance computations (Winter & Depaoli, 2022). The specific set of priors used were as follows.

1) Bayesian estimation with flat (largely **uninformative**) priors, with a normal distribution *N*(0,10,000) for the slope coefficients and uniform(-10,000,10,000) for the variance components;
2) Bayesian estimation with **weak informative fixed effect priors** for the L1 and L2 slope coefficients using *N*(true value,10,000);

3) Bayesian estimation with **weak informative random effect priors** for the L1 and L2 residual variance terms using *IG*(.01,.01) as recommended in previous literature;
4) Bayesian estimation with **strong informative random effect priors** for the L1 and L2 residual variance terms using *IG*(a,b), where a and b are derived from a series of calculated transformations;
5) Combining the weak fixed effect with weak random effect priors (#2 and #3); and finally,
6) Combining the weak fixed effect with strong random effect priors (#2 and #4).

This set of analyses specifically focused on the *IG* distribution to examine the implications of using both weakly and strongly informative priors for fixed effects and random effects. Table 11 documents the estimation time for each analysis approach for both full and subsample sizes. Notably, the Bayesian estimation with the full sample required more time than that with smaller sample. To compute the strong informative variance priors for each sample (full and subsample), I provide a thorough exploration of how chi-square distributed data, often used as a starting point due to its commonality in statistical outputs, can be effectively transformed into gamma (*G*) distributions. The subsequent step to inverse gamma (*IG*) distributions is then detailed, given that *IG* is a common choice for prior distributions in variance parameters due to its conjugacy with normal distributions. The process of determining informative priors for variance components in MLM involved a series of calculated transformations (see Table 12). Below is the process I used for full sample. Figure 15 and Figure 16 show the transformations for the full sample and subsample among chi-square, *G*, and *IG* distribution priors, respectively.

1. Chi-square distribution

- Mean: Calculated based on degrees of freedom (*k*), defined as $n - 1$. For the L2 intercept variance (schools), *k* = 272, and for L1 residual variance (students within schools), *k* = 8422. The simulations set k for the residual variance should be $N - j - p$, not $N - 1$, where N is total size, j is number of clusters, and p is number of fixed predictors at L1. Here, with 2 L1 predictors, k should be $8698 - 273 - 2 = 8423$.
- Mode: For a chi-square distribution is $\max(k - 2, 0)$, yielding modes of 271 for schools and 8421 for students.
- Variance: Defined as $2 \times k$, giving 546 for schools and 16846 for students.

2. Scaling the chi-square distribution

The scaling factor W is calculated as $(s^2 + 2)/k$, where s^2 is the preliminary point estimate of the variance component derived from the maximum likelihood estimate of the 2-level model. For schools, W is approximately to 0.578, and for students, it is 0.254.

3. Gamma (*G*) distribution

- Alpha (a, shape): Half the scaled degrees of freedom, $W \times k/2$.
- Theta (θ, rate): Set at 2 for standardization across transformations.
- Beta (b, scale): Inverse of theta, set at 0.5.

4. Inverse gamma (*IG*) distribution

- Parameters: *IG* (Alpha: $\left(\frac{mean}{var}\right) + 2$, Beta: $mean \times (a - 1)$).
- Derived statistics: mean $b/(a - 1)$, model $b/(a + 1)$, and variance $b^2/((a^2 - 1) * (a - 2))$

The current paper fitted models described in Equation 46 using the *R* brms package (Bürkner, 2017a, b), which is in turn based on the Stan package (Stan Development Team, 2024). The brms package was selected primarily because its model syntax is similar to lme4, which many researchers may be familiar with. Second, brms can be used to fit linear and nonlinear models for unilevel and multilevel data (Barreda & Silbert, 2023). The following section provide the technical description of brms and the methodology behind them (More information about the diverse models that can be fitted with brms and their implications can be found in Bürkner (2017a,b)).

MCMC Estimation in brms

The estimation process in brms uses MCMC sampling, specifically using the No-U-Turn Sampler (NUTS), a variant of Hamiltonian Monte Carlo (HMC). This process involves several steps and checks to ensure the quality of the estimates, as follows.

Number of MCMC chains. By default, brms uses 4 chains for fitting models. This is a common choice to ensure adequate exploration of the parameter space and to facilitate convergence diagnostics for multiple chains. For all analyses, I used this default.

Starting values. The default starting values in brms are based on Stan defaults, which is to randomly select values from a uniform distribution ranging from -2 to +2 (Stan Development Team, 2024). This was also used in my analyses.

Iterations. I also used the default number of iterations per chain, which is 2000. This includes both the burn-in/warm-up period and the iterations from which samples are collected.

Burn-in/warm-up. The burn-in, also known as the warm-up, is a part of the MCMC process where initial samples are discarded. These samples may not represent the stationary distribution of the posterior due to the starting conditions. By default, the first 1000 of the 2000

iterations are used for warm-up. During warm-up, adjustments are made to explore the parameter space more effectively, and these samples are not used for posterior inference. I used the default burn-in setting for analyses.

Thinning. The default thinning rate is set to 1, meaning that every sample after the warm-up is kept to reduce autocorrelation within the sample chain. Again, I used this default.

Posterior Estimates and Credible Intervals

Posterior estimates. After the burn-in period, brms generates samples from the posterior distribution, which are then used to compute summary statistics such as means, medians, and credible intervals. These estimates are derived from the remaining MCMC samples. brms typically provides two forms of posterior estimates: the mean and the median. By default, brms often uses the mean to summarize the central tendency of the posterior distribution by default. This choice can affect the interpretation of the results, especially in skewed distributions where the mean and median might diverge. I used the median for all analyses.

Credible intervals. Credible intervals in brms are used to describe the uncertainty around parameter estimates. By default, brms provides 95% credible intervals, which indicates the range within which 95% of the estimates occur in the posterior. The default in brms (and Mplus) is to report "equal-tailed" intervals (i.e., the range of values from the 2.5^{th} to the 97.5^{th} percentiles). An alternative option is to use the highest posterior probability (HPD) intervals; I used the default for simplicity (but note that using HPD intervals, when I checked using another package, did not change the substantive results of the findings).

Convergence Diagnostics

The brms package provides several diagnostic tools, including the Potential Scale Reduction Factor (PSRF) or R-hat statistic (Bürkner, 2024). The R-hat statistic compares the

variance between multiple chains to the variance within each chain. An R-hat value close to 1 suggests that the chains have converged to the same distribution. Values greater than 1 indicate potential convergence issues, suggesting the need for longer chains or different initial values. (In *Mplus*, a PSRF of 1.05 or less is implemented.)

As already noted, for this model demonstration I used the brms default settings with the important exception of using varied priors for the variance components .

Results

The results from the Bayesian MLM on both the full sample and subsamples are shown in Table 13 and Table 14, and Figure 17 and Figure 18, for the full sample and subsample, respectively.

Influence of Informative Priors on Variance Estimates

Models with uninformative priors consistently showed wider credible intervals for fixed effects and variance components, reflecting greater uncertainty in parameter estimates with less prior knowledge. Introducing weak informative priors using $IG(0.01,0.01)$ slightly narrowed the credible intervals but did not greatly change the variance component estimates. In addition, highly informative priors with $IG(a,b)$ for the subsample yielded much narrower credible intervals, particularly for variance components, demonstrating that stronger priors can notably reduce uncertainty and enhance model stability.

Effect of Informative Priors on Fixed Effects

The estimates of the model fixed effects, L1 intercept, L1 Math effect on Science, L2 Math effect on Science, and their interaction, were relatively stable across different priors used (see Figures 19 – 22 for full sample and Figures 23 – 26 for the subsample). As well, this stability is observed in both the full sample and subsample analyses.

Comparison between Full sample and Subsample Results

The comparison of results between the full sample and the subsample revealed notable differences in the L2 slope coefficient and the L2 variance (in *SD*) component (see again Figures 17 and 18). For the full sample, variability in the L2 parameter estimates was consistent irrespective of prior used, which can be attributed to the very large sample size of schools (i.e., large data overwhelms priors). But for the small sample, there was marked variability among estimates by prior used. Although use of the inverse gamma informative variance prior slightly underestimated the full sample's L2 estimates (red dotted line), the credible interval's upper bound was extremely close to the full sample's value but yet had much more precise interval width. Notably, the Monte Carlo simulations used the true value of the parameter but here in I used the sample-based estimate from the uninformative model.

Overall, the choice of priors had a more pronounced impact on the subsample estimates, highlighting the importance of informative variance priors in smaller sized samples (i.e., multilevel data with few L2 clusters).

CHAPTER TEN:

EQUIVALENCIES AMONG INFORMATIVE VARIANCE PRIORS USING BRMS

Because the Bayesian analysis brms package is not as limited in prior choices for variance as those in Mplus, in this chapter I demonstrate the use of informative variance priors with the previously discussed TIMSS datasets across a wider range of prior distributions, including the gamma (*G*), inverse gamma (*IG*), half-*t*, and half-Cauchy, each of which are appropriate for variance priors given their lower bound of zero, and which were discussed at length in Chapter 3.

Methods

I analyzed the data as described in Chapter 9 using six prior conditions, as follows.

1) Bayesian estimation with flat (largely **uninformative**) priors, with a normal distribution *N*(0,10,000) for the slope coefficients and uniform(0,10,000) for the variance components;
2) Bayesian estimation with **weak informative fixed effect priors** for the L1 slope coefficients using *N*(true value,10,000) and **strong informative random effect priors** for the L1 and L2 residual variance terms using ***G*(a,b)** computed from chi-square drawn from #1;
3) Bayesian estimation with **weak informative fixed effect priors** for the L1 slope coefficients using *N*(true value,10,000) and **strong informative random effect priors** for the L1 and L2 residual variance terms using ***IG*(a,b)**. The parameters a and b of the *IG* distributions are derived from transformations of chi-square distributions to *G* distributions, with initial values drawn from the earlier analysis (referenced as #1);

4) Bayesian estimation with **weak informative fixed effect priors** for the L1 slope coefficients using *N*(true value,10,000) and **weak informative random effect priors** for the L1 and L2 residual variance terms using **Half-*t*(degrees of freedom, location, scale)**, with degrees of freedom (*df*) set to 1. The scale and location parameters are calculated based on earlier results (referenced as #1);
5) Bayesian estimation with **weak informative fixed effect priors** for the L1 slope coefficients using *N*(true value,10,000) and **strong informative random effect priors** for the L1 and L2 residual variance terms using **Half-*t*(*df*, location, scale)**. The *df*, along with the scale and location parameters, are computed based on chi-square values drawn from earlier results (referenced as #1).
6) Bayesian estimation with **weak informative fixed effect priors** for the L1 slope coefficients using *N*(true value,10,000) and **strong informative random effect priors** for the L1 and L2 residual variance terms using **Half-Cauchy(location, scale)**. The distribution is set with a location of 0 and scale parameters derived from chi-square values obtained from previous results (referenced as #1), with $df = 1$.

Table 15 documents the estimation time for each analysis approach, comparing full sample size with subsamples. Consistent with the findings from Chapter 9, Bayesian estimation with the full sample required longer processing time than those required for smaller samples. To compute the strong informative variance priors for both sample types, the initial steps involved transforming chi-square distributed values into *G*, and subsequently into *IG*. Additionally, I included transformations from chi-square to half-*t*, and from half-*t* to half-Cauchy. The transformation methods are described in Table 16, and below I outline the process for the full dataset.

1. Chi-square distribution

- Mean: Calculated based on degrees of freedom (*k*), defined as $n - 1$. For the L2 intercept variance (schools), *k* = 272, and for L1 residual variance (students within schools), *k* = 8422. The simulations set k for the residual variance should be $N - j - p$, not $N - 1$, where N is total size, j is number of clusters, and p is number of fixed predictors at L1. Here, with 2 L1 predictors, k should be $8698 - 273 - 2 = 8423$.
- Mode: For a chi-square distribution is $\max(k - 2, 0)$, yielding modes of 271 for schools and 8421 for students.
- Variance: Defined as $2 \times k$, giving 546 for schools and 16846 for students.

2. Scaling the chi-square distribution

The scaling factor W is calculated as $(s^2 + 2)/k$, where s^2 is the preliminary point estimate of the variance component derived from the maximum likelihood estimate of the 2-level model. For schools, W is approximately to 0.578, and for students, it is 0.254.

3. Gamma (*G*) distribution

- Alpha (a, shape): Half the scaled degrees of freedom, $W \times k/2$.
- Theta (θ, rate): Set at 2 for standardization across transformations.
- Beta (b, scale): Inverse of theta, set at 0.5.

4. Inverse gamma (*IG*) distribution

- Parameters: *IG* (Alpha: $\left(\frac{mean}{var}\right) + 2$, Beta: $mean \times (a - 1)$).
- Derived statistics: mean $b/(a - 1)$, model $b/(a + 1)$, and variance $b^2/((a^2 - 1) * (a - 2))$

5. Half-*t*

- Location = $\sqrt{Oberved\ variance}$
- Scale = $\sqrt{2 \times k \times W}$

6. Half-Cauchy

- A special case of the half-*t* distribution with one degree of freedom, that is, half-Cauchy (0, scale), where scale = $\sqrt{2 \times k \times W}$.

Results

Influence of Informative Priors on Variance Estimates

Tables 17 and 18 demonstrated the results, while Figures 27 and 28 provide a visualization of the results across different prior settings for the full sample and subsample, respectively. As can be seen, the results indicate informative priors greatly reduce the variation in estimated parameters compared to uninformative priors. As in the previous results, the use of informative priors had little or no impact for the full sample model results, likely because the sample size is so large that it overwhelms the priors. However, as observed in Figure 28, for the subsample, the L2 variance component (in *SD*) was best estimated by gamma and inverse gamma, rather than half-t and half-Cauchy, as indicated by the width of the credible interval.

Effect of Informative Priors on Fixed Effects

The estimates of fixed effects are shown in Figures 29 – 32 for the full sample and Figures 33 – 36 for the subsample, respectively. Just as was observed in the previous results, both the full sample and subsample estimates were consistent irrespective of prior used.

CHAPTER ELEVEN:

DISCUSSION

This dissertation investigated the impact of the use of informative variance priors in Bayesian multilevel modeling (MLMs), with a focus on a 2-level hierarchical random intercept linear regression model. The motivation behind this study stems from the challenges in modeling complex data structures, especially when the number of higher-level (L2) units is limited, which is a common scenario in educational and social behavioral research. Understanding how to use informative priors are important in such settings because they can enhance the accuracy and credibility of the estimates, thus providing more reliable and interpretable results for researchers and practitioners (Ulitzsch et al., 2021), particularly for smaller samples.

To date, Bayesian default variance priors differ among available software, most of which use defaults that are uninformative or weakly informative. Further, results from previous research are not unified in their recommendations for practitioners (Baldwin & Fellingham, 2013; Gelman, 2006; McNeish & Stapleton, 2016a,b). This dissertation seeks to clarify the impact of the use of informative variance component priors on estimation accuracy and coefficient credible intervals in MLMs that are estimated using Bayesian MCMC. My particular focus was on L2 intercept variance in two-level hierarchical models, with an eye toward building upon this work in several ways, such as examining priors for slope variances and intercept-slope covariances as well as more complex model structures.

One of the key aspects of this dissertation was to demonstrate the transformability among different distributions used for setting informative priors, specifically the chi-square, gamma (*G*), inverse gamma (*IG*), half-*t*, and half-Cauchy distributions. These distributions are important in the context of variance components in MLMs because they naturally align with the theoretical

distributions of variance estimates. As part of this dissertation, I detailed how hyperparameters from one of these distributions can be transformed into those for another, providing a methodological bridge that enhances the flexibility and applicability of prior specification. This transformability is essential for adapting the priors to the specific needs and characteristics of the data being analyzed.

The results from the Monte Carlo simulations highlighted the effectiveness of using informative variance priors across a variety of MLM conditions. These simulations varied the number of clusters, cluster sizes, ICCs, and the proportion of variance explained at both levels of the model. The findings clearly showed that informative priors help in reducing bias and improving the precision of the variance component estimates, particularly in scenarios with small sample sizes or high ICCs. Models with appropriately chosen informative priors were consistently superior in terms of false positive rates and bias compared to models with uninformative or weakly informative priors.

Nevertheless, our findings indicate two caveats. First, the gamma prior specification we used in the simulation as well as the applied demonstration may be overly wide (we used a b parameter of 2, but it is possible to recalibrate the hyperparameters to use a b parameter of 3 instead). Future work will examine this option. Second, using the inverse gamma prior with a known population value resulted in slight undercoverage, which may be why the applied subsample analysis using an informative inverse gamma prior did not fully capture the larger sample's variance component estimate (see again Figures 18 and 28).

Overall, the applied analysis using data from the TIMSS 2019 Grade 8 U.S. Science achievement datasets further substantiated the simulation results. This real-data demonstration emphasized the practical benefits of employing Bayesian informative priors in actual research

scenarios. In particular, the applied analysis highlighted how inverse gamma variance priors can enhance obtaining precise variance estimates for smaller sized samples. Specifically, the inverse gamma prior had narrower intervals compared to gamma, half-*t*, and half-Cauchy. This confirms that the specific shape and scale parameters used in the inverse gamma distribution are particularly well-suited to capturing the underlying variance structure in smaller sized samples.

In part, this dissertation also demonstrates that the risks associated with poor selection of priors can be substantial. If parameter prior settings are too strict, the posterior estimates may be biased downward; conversely, if the priors for parameters are set too loosely, the posterior distribution may be too wide. Considering these issues, it is essential to conduct sensitivity analyses with priors. This process involves systematically varying the priors to observe how sensitive the model results are across different settings. This kind of analysis aims to help identify ranges of parameter values that are consistent with empirical data and theoretical expectations and assess the robustness of model to the priors.

Limitations and Conclusion

Like any study, this dissertation was focused on particular data conditions, model formulations, and analytic choices. For example, we used a relatively simple 2-level linear model with one random effect (intercept variance component). Many other models exist, including Bernoulli and Poisson, for example. Nevertheless, we doubt that the variance distributions would differ much for nonlinear models compared to linear models, since the L2 variance component is often assumed to still vary in a chi-square fashion. As well, I used *Mplus* and *R* brms for all analyses and did not change default Bayesian analysis settings within them. This said, there is no reason to believe that the results would be meaningfully different using other algorithms or settings. Last, I did not systematically investigate the use of a 2-step process for identifying prior

hyperparameters, particularly for small samples (e.g., Winter & Depaoli, 2022). This is certainly an important area for future research to consider.

In conclusion, this dissertation set out to provide: 1) use of informative variance priors based on principled evidence rather than arbitrary rules of thumb, but without sacrificing accuracy, 2) a review of available software with defaults for implementing Bayesian analyses, and 3) a concrete method for implementing a number of potential variance prior distributions for use in Bayesian analysis, including chi-square, *G*, *IG*, half-*t*, and half-Cauchy. Through both simulation and demonstration, my results point to valuable guidance for applied researchers around best practices in using informative variance priors in Bayesian multilevel models. Last, this work aimed also to advance improved clarity around the usefulness of Bayesian analyses so that researchers can develop better understandings of their data for more informed policymaking.

REFERENCES

Araya, R., Zitko, P., Markkula, N., Rai, D., & Jones, K. (2018). Determinants of access to health care for depression in 49 countries: A multilevel analysis. *Journal of Affective Disorders*, *234*, 80–88. https://doi.org/10.1016/j.jad.2018.02.092

Asparouhov, T., & Muthén, B. (2010a). Bayesian analysis of latent variable models using Mplus. Retrieved from: https://www.statmodel.com/download/BayesAdvantages18.pdf

Asparouhov, T., & Muthén, B. (2010b). Bayesian analysis using Mplus: Technical implementation. Retrieved from: https://statmodel.com/download/Bayes3.pdf

Baek, E., Beretvas, S. N., Van den Noortgate, W., & Ferron, J. M. (2020). Brief research report: Bayesian versus REML estimations with noninformative priors in multilevel single-case data. *The Journal of Experimental Education*, *88*(4), 698-710. https://doi.org/10.1080/00220973.2018.1527280

Baek, E., & Ferron, J. J. (2020). Modeling heterogeneity of the level-1 error covariance matrix in multilevel models for single-case data. *Methodology*, *16*(2), 166-185

Baldwin, S. A., & Fellingham, G. W. (2013). Bayesian methods for the analysis of small sample multilevel data with a complex variance structure. *Psychological Methods*, *18*(2), 151. https://doi.org/10.1037/a0030642

Balakrishnan, N., & Nevzorov, V. B. (2003). *A primer on statistical distributions*. John Wiley & Sons.

Bandyopadhyay, P. S., & Cherry, S. (2011). Elementary probability and statistics: A primer. In *Philosophy of Statistics* (pp. 53-95). North-Holland.

Barreda, S., & Silbert, N. (2023). Fitting Bayesian regression models with brms. In *Bayesian Multilevel Models for Repeated Measures Data* (pp. 55-86). Routledge.

Betancourt, M. (2017). *How the Shape of a Weakly Informative Prior Affects Inferences*. Available online at: https://mc-stan.org/users/documentation/case-studies/weakly_informative_shapes.html#:~:text=When%20the%20scales%20are%20well,how%20strong%20that%20regularization%20is.

Betancourt, M. (2019). The Convergence of Markov Chain Monte Carlo Methods: From the Metropolis Method to Hamiltonian Monte Carlo. *Annalen Der Physik, 531*(3). https://doi.org/10.1002/andp.201700214

Betancourt, M., & Girolami, M. (2015). Hamiltonian Monte Carlo for hierarchical models. *Current trends in Bayesian methodology with applications*, *79*(30), 2-4.

Bhaumik, D. K., Kapur, K., & Gibbons, R. D. (2009). Testing parameters of a gamma distribution for small samples. *Technometrics*, *51*(3), 326-334.

Bolin, J. H., Finch, W. H., & Stenger, R. (2019). Estimation of random coefficient multilevel models in the context of small numbers of level 2 clusters. *Educational and Psychological Measurement, 79*(2), 217–248. https://doi.org/10.1177/0013164418773494

Boomsma, A. (1982). Systems under indirect observation: Causality, structure, prediction. *The robustness of lisrel against small sample sizes in factor analysis models*, 149-173.

Boomsma, A. (1985). Nonconvergence, improper solutions, and starting values in LISREL maximum likelihood estimation. *Psychometrika*, *50*, 229-242.

Bressler, D. M., Shane Tutwiler, M., & Bodzin, A. M. (2021). Promoting student flow and interest in a science learning game: a design-based research study of School Scene Investigators. *Educational Technology Research and Development*, *69*(5), 2789–2811. https://doi.org/10.1007/s11423-021-10039-y

Browne, W. J., & Draper, D. (2000). Implementation and performance issues in the Bayesian and likelihood fitting of multilevel models. *Computational Statistics*, *15*(3), 391–420. https://doi.org/10.1007/s001800000041

Browne, W. J., & Draper, D. (2006). A comparison of Bayesian and likelihood-based methods for fitting multilevel models. *Bayesian Analysis*, *1*(3). https://doi.org/10.1214/06-BA117

Browne, W. J., & Rasbash, J. (2009). *MCMC estimation in MLwiN*. Bristol: Centre of Multilevel Modelling, University of Bristol.

Buckley, J., & Schneider, M. (2005). Are charter school students harder to educate? Evidence from Washington, DC. *Educational Evaluation and Policy Analysis*, *27*(4), 365-380. https://doi.org/10.14507/epaa.v13n22.2005

Bürkner, P. C. (2017a). Advanced Bayesian multilevel modeling with the R package brms. *arXiv preprint arXiv:1705.11123*.

Bürkner, P. C. (2017b). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*, 1-28.

Bürkner, P. C. (2024). *Estimating Distributional Models with brms.* Retrieved from: https://cran.r-project.org/web/packages/brms/vignettes/brms_distreg.html

Carlin, B. P., & Chib, S. (1995). Bayesian model choice via Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *57*(3), 473-484. https://doi.org/10.1111/j.2517-6161.1995.tb02042.x

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P., & Riddell, A. (2017). Stan: A Probabilistic Programming Language. *Journal of Statistical Software*, *76*(1), 1–32. https://doi.org/10.18637/jss.v076.i01

Casella, G., & Berger, R. L. (2002). *Statistical inference* (2nd ed.). Thomson Learning.

Chance, B., del Mas, R., & Garfield, J. (2004). Reasoning about sampling distribitions. In *The challenge of developing statistical literacy, reasoning and thinking* (pp. 295-323). Dordrecht: Springer Netherlands.

Chang, H. J., Wu, C. H., Ho, J. F., & Chen, P. Y. (2008). On sample size in using central limit theorem for gamma distribution. *International Journal of Information and Management sciences*, *19*(1), 153-174.

Chen, T., Fox, E., & Guestrin, C. (2014). Stochastic gradient Hamiltonian Monte Carlo. In *Proceedings of the 31st International Conference on Machine Learning, Proceedings of Machine Learning Research, 32*(2), 1683-1691.

Claessens, A., & Engel, M. (2013). How important is where you start? Early mathematics knowledge and later school success. *Teachers College Record, 115*(6), 1-29. https://eric.ed.gov

Conrady, S., & Jouffe, L. (2015). *Bayesian networks and BayesiaLab: a practical introduction for researchers* (Vol. 9). Franklin: Bayesia USA.

Cook, J. D. (2008). *Inverse Gamma Distribution*. Retrieved from: https://www.johndcook.com/inverse_gamma.pdf.

Daniels, M. J. (1999). A prior for the variance in hierarchical models. *Canadian Journal of Statistics*, *27*(3), 567-578. https://doi.org/10.2307/3316112

Denwood, M. J. (2016). runjags: An R Package Providing Interface Utilities, Model Templates, Parallel Computing Methods and Additional Distributions for MCMC Models in JAGS. *Journal of Statistical Software*, *71*(9), 1–25. https://doi.org/10.18637/jss.v071.i09

Depaoli, S., & Clifton, J. P. (2015). A Bayesian approach to multilevel structural equation modeling with continuous and dichotomous outcomes. *Structural Equation Modeling: A Multidisciplinary Journal*, *22*(3), 327-351. https://doi.org/10.1080/10705511.2014.937849

Doyle, W. R. (2010). US Senator's Ideal Points for Higher Education: Documenting Partisanship, 1965–2004. *The Journal of Higher Education*, *81*(5), 619-644. https://doi.org/10.1080/00221546.2010.11779069

Doyle, W. R., & Gorbunov, A. V. (2011). The Growth of Community Colleges in the American States: An Application of Count Models to Institutional Growth. *Teachers College Record (1970)*, *113*(8), 1794–1826. https://doi.org/10.1177/016146811111300808

Farrell, S., & Ludwig, C. J. H. (2008). Bayesian and maximum likelihood estimation of hierarchical response time models. *Psychonomic Bulletin & Review*, *15*(6), 1209–1217. https://doi.org/10.3758/PBR.15.6.1209

Finch, W. H., Bolin, J. E., & Kelley, K. (2019). *Multilevel modeling using R* (Second, Ser. Statistics in the social and behavioral sciences series). CRC Press/Taylor & Francis.

Fishbein, B., Foy, P., & Yin, L. (2021). *TIMSS 2019 user guide for the international database.* Retrieved from https://timssandpirls.bc.edu/timss2019/international-database

Forbes, C., Evans, M., Hastings, N., & Peacock, B. (2011). *Statistical distributions*. John Wiley & Sons.

Freund, R. J., Wilson, W. J., & Mohr, D. L. (2010). *Statistical methods, students solutions manual (e-only)*. Academic Press.

Gabry, J., & Goodrich, B. (2020). *Prior Distributions for rstanarm Models. https://mc-stan.org/rstanarm/articles/priors.html*

Galbraith, C. S., & Merrill, G. B. (2012). Faculty Research Productivity and Standardized Student Learning Outcomes in a University Teaching Environment: A Bayesian Analysis of Relationships. *Studies in Higher Education (Dorchester-on-Thames)*, *37*(4), 469–480. https://doi.org/10.1080/03075079.2010.523782

Galbraith, C. S., Merrill, G. B., & Kline, D. M. (2012). Are Student Evaluations of Teaching Effectiveness Valid for Measuring Student Learning Outcomes in Business Related Classes? A Neural Network and Bayesian Analyses. *Research in Higher Education*, *53*(3), 353–374. https://doi.org/10.1007/s11162-011-9229-0

Ge, H., Xu, K., & Ghahramani, Z. (2018). Turing: a language for flexible probabilistic inference. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, *Proceedings of Machine Learning Research, 84*, 1682-1690.

Gelfand, A. E., Hills, S. E., Racine-Poon, A., & Smith, A. F. (1990). Illustration of Bayesian inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association*, *85*(412), 972-985. https://doi.org/10.1080/01621459.1990.10474968

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*. Chapman and Hall/CRC.

Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (Comment on Article by Browne and Draper). *Bayesian Analysis.* 1(3), 515–534. https://doi.org/10.1214/06-ba117a

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian Data Analysis*, 3rd Edn. CRC Press.

Ghosh, P., & Albert, P. S. (2009). A Bayesian analysis for longitudinal semicontinuous data with an application to an acupuncture clinical trial. *Computational Statistics and Data Analysis*, *53*(3), 699–706. https://doi.org/10.1016/j.csda.2008.09.011

Goodrich B, Gabry J, Ali I & Brilleman S. (2020). *rstanarm: Bayesian applied regression modeling via Stan*. (Version 2.21.1) [Software]. Retrieved from: https://mc-stan.org/rstanarm.

Goudie, R. J. B., Turner, R. M., De Angelis, D., & Thomas, A. (2020). MultiBUGS: A Parallel Implementation of the BUGS Modelling Framework for Faster Bayesian Inference. *Journal of Statistical Software*, *95*, 7. https://doi.org/10.18637/jss.v095.i07

Gudmestad, A., House, L., & Geeslin, K. L. (2013). What a Bayesian Analysis Can Do for SLA: New Tools for the Sociolinguistic Study of Subject Expression in L2 Spanish. *Language Learning*, *63*(3), 371–399. https://doi.org/10.1111/lang.12006

Gustafson, P., Hossain, S., & Macnab, Y. C. (2006). Conservative prior distributions for variance parameters in hierarchical models. *Canadian Journal of Statistics*, *34*(3), 377-390.

Guo, J., Gabry, J., Goodrich, B., & Weber, S. (2020). *Package 'rstan'*. Retrieved from: https://cran.r-project.org/web/packages/rstan/index.html.

Hallquist, M. N., & Wiley, J. F. (2018). MplusAutomation: An R Package for Facilitating Large-Scale Latent Variable Analyses in Mplus. *Structural Equation Modeling: A Multidisciplinary Journal*, *25*(4), 621–638. https://doi.org/10.1080/10705511.2017.1402334

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (Vol. 2, pp. 1-758). Springer.

Heck, R., & Thomas, S. L. (2020). *An introduction to multilevel modeling techniques: MLM and SEM approaches*. Routledge.

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. https://doi.org/10.3102/0162373707299706

Hoffman, M. D., & Gelman, A. (2014). The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, *15*(1), 1593-1623.

Hogg, R. V., McKean, J. W., & Craig, A. T. (2013). *Introduction to mathematical statistics*. Pearson Education India.

Holtmann, J., Koch, T., Lochner, K., & Eid, M. (2016). A comparison of ML, WLSMV, and Bayesian methods for multilevel structural equation models in small samples: A simulation study. *Multivariate Behavioral Research*, *51*(5), 661-680. https://doi.org/10.1080/00273171.2016.1208074

Hoogerheide, L., Block, J. H., & Thurik, R. (2012). Family background variables as instruments for education in income regressions: A Bayesian analysis. *Economics of Education Review*, *31*(5), 515–523. https://doi.org/10.1016/j.econedurev.2012.03.001

Hoogland, J. J., & Boomsma, A. (1998). Robustness Studies in Covariance Structure Modeling: An Overview and a Meta-Analysis. *Sociological Methods & Research*, *26*(3), 329–367. https://doi.org/10.1177/0049124198026003003

Hox, J., & McNeish, D. (2020). Small samples in multilevel modeling. *Small Sample Size Solutions*, 215-225. https://doi.org/10.4324/9780429273872-18

Hox, J., & Roberts, J. K. (Eds.). (2011). *Handbook of advanced multilevel analysis*. Psychology Press.

Illowsky, B., & Dean, S. (2018). *Introductory Statistics.* Open Access Textbooks. *6.* https://commons.erau.edu/oer-textbook/6

Jephcote, C., Medland, E., & Lygo-Baker, S. (2021). Grade inflation versus grade improvement: Are our students getting more intelligent? *Assessment and Evaluation in Higher Education*, *46*(4), 547–571. https://doi.org/10.1080/02602938.2020.1795617

Jackman, S. (2009). *Bayesian analysis for the social sciences*. John Wiley & Sons.

Jin, H., & Rubin, D. B. (2009). Public Schools versus Private Schools: Causal Inference with Partial Compliance. *Journal of Educational and Behavioral Statistics*, *34*(1), 24–45. https://doi.org/10.3102/1076998607307475

Johnson, N. L., Kotz, S., & Balakrishnan, N. (1995). *Continuous univariate distributions, volume 2* (Vol. 289). John Wiley & Sons.

Kaplan, D. (2019). Bayesian Inference for Social Policy Research. *OPRE Report*, *36*. Retrieved from: https://acfmainstage.acf.hhs.gov/sites/default/files/documents/opre/opre_kaplanbrief_042319_508_1.pdf

Kaplan, D. (2023). *Bayesian statistics for the social sciences*. Guilford Publications.

Kaplan, D. (2023). Common probability distributions and their priors. In D. Kaplan, *Bayesian statistics for the social sciences* (pp. 31-46). Guilford Publications.

Kéry, M., & Schaub, M. (2011). *Bayesian population analysis using WinBUGS: a hierarchical perspective*. Academic Press.

Kissell, R., & Poserina, J. (2017). *Optimal sports math, statistics, and fantasy*. Academic Press.

Koch, K.-R. (2007). *Introduction to Bayesian Statistics.* Springer-Verlag Berlin Heidelberg.

Kolovou, D., Naumann, A., Hochweber, J., & Praetorius, A.-K. (2021). Content-specificity of teachers' judgment accuracy regarding students' academic achievement. *Teaching and Teacher Education, 100,* 103298-. https://doi.org/10.1016/j.tate.2021.103298

Kruschke, J. (2014). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press.

Kumar, R., Carroll, C., Hartikainen, A., & Martin, O. (2019). ArviZ a unified library for exploratory analysis of Bayesian models in Python. *Journal of Open Source Software, 4*(33), 1143-. https://doi.org/10.21105/joss.01143

Lee, J., Son, J., Zhou, S., & Chen, Y. (2020). Variation source identification in manufacturing processes using Bayesian approach with sparse variance components prior. *IEEE Transactions on Automation Science and Engineering*, *17*(3), 1469-1485.

Lee, S. Y., & Song, X. Y. (2004). Evaluation of the Bayesian and maximum likelihood approaches in analyzing structural equation models with small sample sizes. *Multivariate Behavioral Research*, *39*(4), 653-686. https://doi.org/10.1207/s15327906mbr3904_4

Lemoine, N. P. (2019). Moving beyond noninformative priors: why and how to choose weakly informative priors in Bayesian analyses. *Oikos*, *128*(7), 912-928. https://doi.org/10.1111/oik.05985

Levine, M. (2020). *Lecture 7: Uniform, Exponential, Gamma, Inverse Gamma, and Beta Distributions*. STAT 516: Some Special Continuous Distributions. Purdue University. Retrieved from: https://www.stat.purdue.edu/~mlevins/docs/stat516/lecture_notes/ContinuousDistr.pdf

Levy, R. (2024). *Bayesian Statistical Modeling: A Second Course: Factor Analysis* [Online course].

Levy, R., & McNeish, D. (2021). Perspectives on Bayesian inference and their implications for data analysis. *Psychological Methods*. https://doi.org/10.1037/met0000443

Liang, X. (2020). Prior sensitivity in Bayesian structural equation modeling for sparse factor loading structures. *Educational and Psychological Measurement, 80*(6), 1025-1058. https://doi.org/10.1177/0013164420906449

Lipscomb, S. T., Miao, A. J., Finders, J. K., Hatfield, B., Kothari, B. H., & Pears, K. (2019). Community-Level Social Determinants and Children's School Readiness. *Prevention Science, 20*(4), 468–477. https://doi.org/10.1007/s11121-019-01002-8

Llera, A., & Beckmann, C. F. (2016). Estimating an inverse gamma distribution. *arXiv preprint arXiv:1605.01019*.

Lunn, D., Spiegelhalter, D., Thomas, A., & Best, N. (2009). The BUGS project: Evolution, critique and future directions. *Statistics in Medicine*, *28*(25), 3049-3067. https://doi.org/10.1002/sim.3680

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS-a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, *10*, 325-337. https://doi.org/10.1023/A:1008929526011

Luo, W., Li, H., Baek, E., Chen, S., Lam, K. H., & Semma, B. (2021). Reporting practice in multilevel modeling: A revisit after 10 years. *Review of Educational Research*, *91*(3), 311-355. https://doi.org/10.3102/003465432199122

Ma, D. (2017). *Transformed gamma distribution*. Topics in Actuarial Modeling. https://actuarialmodelingtopics.wordpress.com/2017/06/17/transformed-gamma-distribution/

Mai, Y., & Zhang, Z. (2018). Software Packages for Bayesian Multilevel Modeling. *Structural Equation Modeling: A Multidisciplinary Journal*, *25*(4), 650–658. https://doi.org/10.1080/10705511.2018.1431545

Martin, O. (2018). *Bayesian analysis with Python: introduction to statistical modeling and probabilistic programming using PyMC3 and ArviZ* (Second edition.). Packt Publishing.

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.

McNeish, D. (2016a). On using Bayesian methods to address small sample problems. *Structural Equation Modeling: A Multidisciplinary Journal*, *23*(5), 750–773. https://doi.org/10.1080/10705511.2016.1186549

McNeish, D. (2016b). Using data-dependent priors to mitigate small sample bias in latent growth models: A discussion and illustration using Mplus. *Journal of Educational and Behavioral Statistics*, *41*(1), 27–56. https://doi.org/10.3102/1076998615621299

McNeish, D., & Stapleton, L. M. (2016a). Modeling clustered data with very few clusters. *Multivariate Behavioral Research*, *51*(4), 495-518. https://doi.org/10.1080/00273171.2016.1167008

McNeish, D. M., & Stapleton, L. M. (2016b). The effect of small sample size on two-level model estimates: A review and illustration. *Educational Psychology Review*, *28*, 295-314. https://doi.org/10.1007/s10648-014-9287-x

McNeish, D. (2019). Poisson multilevel models with small samples. *Multivariate Behavioral Research*, *54*(3), 444-455. https://doi.org/10.1080/00273171.2018.1545630

Moeyaert, M., Rindskopf, D., Onghena, P., & Van den Noortgate, W. (2017). Multilevel modeling of single-case data: A comparison of maximum likelihood and Bayesian estimation. *Psychological Methods*, *22*(4), 760. https://doi.org/10.1037/met0000136

Monnahan, C. C., & Kristensen, K. (2018). No-U-turn sampling for fast Bayesian inference in ADMB and TMB: Introducing the adnuts and tmbstan R packages. *PloS one*, *13*(5), e0197954. https://doi.org/10.1371/journal.pone.0197954

Monnahan, C. C., Muradian, M. L., & Kuriyama, P. T. (2014). A guide for Bayesian analysis in AD Model Builder. https://www.admb-project.org/developers/mcmc/mcmc-guide-for-admb.pdf

Mood, A. M., Graybill, F. A., & Boes, D. C. (1973). *Introduction to the Theory of Statistics*. McGraw-Hill Science, Engineering & Mathematics.

MSU Libraries. (n.d.). *Regularized gamma function*. Retrieved from: https://archive.lib.msu.edu/crcmath/math/math/r/r200.htm

Mullis, I. V., Martin, M. O., Foy, P., Kelly, D. L., & Fishbein, B. (2020). TIMSS 2019 international results in mathematics and science. *Retrieved from Boston College, TIMSS & PIRLS International Study Center website: https://timssandpirls. bc. edu/timss2019/international-results*.

Muthén, L.K. and Muthén, B.O. (1998-2017). *Mplus User's Guide*. Eighth Edition. Los Angeles, CA: Muthén & Muthén

Nalborczyk, L., Batailler, C., Lœvenbruck, H., Vilain, A., & Bürkner, P. C. (2019). An introduction to Bayesian multilevel models using brms: A case study of gender effects on vowel variability in standard Indonesian. *Journal of Speech, Language, and Hearing Research*, *62*(5), 1225-1242. https://doi.org/10.1044/2018_JSLHR-S-18-0006

Niemi, J. (2021). *Inverse gamma distribution*. Retrieved from: https://www.jarad.me/courses/stat587Eng/slides/Probability/Distributions/Inverse_gamma/Inverse_gamma.pdf

Notenboom, A., & Reitsma, P. (2007). Spelling Dutch doublets: Children's learning of a phonological and morphological spelling rule. *Scientific Studies of Reading*, *11*(2), 133-150. https://doi.org/10.1080/10888430709336556

Ntzoufras, I. (2011). *Bayesian modeling using WinBUGS*. John Wiley & Sons.

OpenAI. (2023). *ChatGPT* (Mar 14 version) [Large language model]. https://chat.openai.com/chat

Papoulis, A. (1965). *Random variables and stochastic processes*. McGraw Hill.

Penn State. (n.d.). *16.5 - The Standard Normal and The Chi-Square.* In STAT 414: *Introduction to Probability Theory*. Pennsylvania State University. Retrieved from: https://online.stat.psu.edu/stat414/lesson/16/16.5

Penn State. (n.d.). *26.3 - Sampling Distribution of Sample Variance* In *STAT 414*: *Introduction to Probability Theory*. Pennsylvania State University. Retrieved from: https://online.stat.psu.edu/stat414/lesson/26/26.3

Petrykowski, K. (2020, October 12). *Chi-squared distribution is a special case of gamma distribution*. The Book of Statistical Proofs. https://statproofbook.github.io/P/chi2-gam.html

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In K Hornik, F Leisch, A Zeileis (eds.), *Proceedings of the 3rd international workshop on distributed statistical computing* (Vol. 124, No. 125.10, pp. 1-10). https://www.R-project.org/conferences/DSC-2003/Proceedings/Plummer.pdf.

Plummer, M., Best, N., Cowles, K., & Vines, K. (2006). CODA: convergence diagnosis and output analysis for MCMC. *R news*, *6*(1), 7-11.

Polson, N. G., & Scott, J. G. (2012). On the half-Cauchy prior for a global scale parameter. *Bayesian Analysis*, *7*(4), 887-902. http://dx.doi.org/10.1214/12-BA730

Probability Distribution Explorer (n.d.). *Half-Cauchy distribution*. https://distribution-explorer.github.io/continuous/halfcauchy.html

Psarakis, S., & Panaretoes, J. (1990). The folded t distribution. *Communications in Statistics-Theory and Methods*, *19*(7), 2717-2734. https://doi.org/10.1080/03610929008830342

Ramachandran, K. M., & Tsokos, C. P. (2021). Chapter 4 - Sampling distributions. In K. M. Ramachandran & C. P. Tsokos (Eds.), *Mathematical statistics with applications in R* (Third edition, pp. 147-177). Academic Press. https://doi.org/10.1016/B978-0-12-817815-7.00004-X

Raudenbush, S. W., & Bryk, A. S. (2002). *Hierarchical linear models: applications and data analysis methods* (2nd ed.). Sage Publications.

R Core Team. (2023). *R: A language and environment for statistical computing* [Software]. R Foundation for Statistical Computing. https://www.R-project.org/

Rodríguez, I., Kozusznik, M. W., Peiró, J. M., & Tordera, N. (2019). Individual, co-active and collective coping and organizational stress: A longitudinal study. *European Management Journal*, *37*(1), 86–98. https://doi.org/10.1016/j.emj.2018.06.002

Ross, S. M. (2014a). Chapter 6 - Distributions of Sampling Statistics. In Sheldon M. Ross (Ed.), *Introduction to Probability and Statistics for Engineers and Scientists (Fifth Edition)* (pp. 207-233). Academic Press. https://doi.org/10.1016/B978-0-12-394811-3.50006-X.

Ross, S. M. (2014b). *Introduction to probability models*. Academic press.

Rue, H., Martino, S., & Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *71*(2), 319-392. https://doi.org/10.1111/j.1467-9868.2008.00700.x

Ruf, A., Neubauer, A. B., Ebner-Priemer, U., Reif, A., & Matura, S. (2021). Studying dietary intake in daily life through multilevel two-part modelling: a novel analytical approach and its practical application. *International Journal of Behavioral Nutrition and Physical Activity*, *18*(1), 1-14. https://doi.org/10.1186/s12966-021-01187-8

Rupp, A. A., Dey, D. K., & Zumbo, B. D. (2004). To Bayes or not to Bayes, from whether to when: Applications of Bayesian methodology to modeling. *Structural Equation Modeling: A Multidisciplinary Journal*, *11*(3), 424-451. https://doi.org/10.1207/s15328007sem1103_7

Salvatier, J., Wiecki, T. V., & Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, *2*, e55.

SAS Institute Inc. (2020). *PRIOR statement*. In *SAS/STAT® 15.2 User's Guide*. Retrieved from https://documentation.sas.com/doc/en/statug/15.2/statug_mixed_syntax12.htm

Scarpino, S. E., Lawrence, F. R., Davison, M. D., & Hammer, C. S. (2011). Predicting bilingual Spanish-English children's phonological awareness abilities from their preschool English and Spanish oral language. *Journal of Research in Reading*, *34*(1), 77–93. https://doi.org/10.1111/j.1467-9817.2010.01488.x

Shreffler, J., & Huecker, M. R. (2022). Hypothesis testing, P values, confidence intervals, and significance. In *StatPearls [Internet]*. StatPearls Publishing. Available from: https://www.ncbi.nlm.nih.gov/books/NBK557421/

Singh, K., Granville, M., & Dika, S. (2002). Mathematics and science achievement: Effects of motivation, interest, and academic engagement. *The Journal of Educational Research, 95*(6), 323–332. https://doi.org/10.1080/00220670209596607

Skinner, B., Burtch, T., & Levy, H. (2024). Variation in Broadband Access Among Undergraduate Populations Across the United States. *Research in Higher Education*. https://doi.org/10.1007/s11162-024-09775-w

Skinner, B. T., & Doyle, W. R. (2024). Predicting postsecondary attendance by family income in the United States using multilevel regression with poststratification. *Economics of Education Review*, *99*, 102508. https://doi.org/10.1016/j.econedurev.2024.102508

Smid, S. C., Depaoli, S., & Van De Schoot, R. (2020a). Predicting a distal outcome variable from a latent growth model: ML versus Bayesian estimation. *Structural Equation Modeling: A Multidisciplinary Journal*, *27*(2), 169-191. https://doi.org/10.1080/10705511.2019.1604140

Smid, S. C., McNeish, D., Miočević, M., & van de Schoot, R. (2020b). Bayesian versus frequentist estimation for structural equation models in small sample contexts: A systematic review. *Structural Equation Modeling: A Multidisciplinary Journal*, *27*(1), 131-161. https://doi.org/10.1080/10705511.2019.1577140

Smid, S. C., & Winter, S. D. (2020). Dangers of the defaults: A tutorial on the impact of default priors when using Bayesian SEM with small samples. *Frontiers in Psychology*, *11*, 611963. https://doi.org/10.3389/fpsyg.2020.611963

Snijders, T. A., & Bosker, R. J. (2011). *Multilevel analysis: An introduction to basic and advanced multilevel modeling*. Sage.

Soch, J., et al. (2024). *StatProofBook/StatProofBook.github.io: The Book of Statistical Proofs* (Version 2023). Zenodo. https://doi.org/10.5281/ZENODO.4305949

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, *64*(4), 583–639. https://doi.org/10.1111/1467-9868.00353

Stan Development Team. (2024). *Stan: A C++ Library for Probability and Sampling, Version 2.35.0* [Software]. Available from https://mc-stan.org

Stapleton, L. M., McNeish, D. M., & Yang, J. S. (2016a). Multilevel and Single-Level Models for Measured and Latent Variables When Data Are Clustered. *Educational Psychologist*, *51*(3–4), 317–330. https://doi.org/10.1080/00461520.2016.12071781

Stapleton, L. M., Yang, J. S., & Hancock, G. R. (2016b). Construct meaning in multilevel settings. *Journal of Educational and Behavioral Statistics*, *41*(5), 481-520. https://doi.org/10.3102/1076998616646200

Stegmueller, D. (2013). How Many Countries for Multilevel Modeling? A Comparison of Frequentist and Bayesian Approaches. *American Journal of Political Science*, *57*(3), 748–761. https://doi.org/10.1111/ajps.12001

Stokes, M., Chen, F., & Gunes, F. (2014). An introduction to Bayesian analysis with SAS/STAT® software. In *Proceedings of the SAS Global Forum 2014 Conference, SAS Institute Inc, Cary, USA* (available at https://support.sas.com/resources/papers/proceedings14/SAS400-2014.pdf).

Stuart, A., & Ord, K. (2010). *Kendall's advanced theory of statistics, distribution theory* (Vol. 1). John Wiley & Sons.

Tafvelin, S., & Stenling, A. (2021). A Self-Determination Theory Perspective on Transfer of Leadership Training: The Role of Leader Motivation. *Journal of Leadership & Organizational Studies*, *28*(1), 60–75. https://doi.org/10.1177/1548051820962504

Tamminen, K. A., Gaudreau, P., McEwen, C. E., & Crocker, P. R. E. (2016). Interpersonal Emotion Regulation Among Adolescent Athletes: A Bayesian Multilevel Model Predicting Sport Enjoyment and Commitment. *Journal of Sport & Exercise Psychology*, *38*(6), 541–555. https://doi.org/10.1123/jsep.2015-0189

Thomas, A., O'Hara, B., Ligges, U., & Sturtz, S. (2006). Making BUGS Open. *R News*, 6(1), 12–17. https://cran.r-project.org/doc/Rnews/.

Tiao, G. C., & Box, G. E. (1973). Some comments on "Bayes" estimators. *The American Statistician*, *27*(1), 12-14.

Ulitzsch, E., Lüdtke, O., & Robitzsch, A. (2023). Alleviating Estimation Problems in Small Sample Structural Equation Modeling-A Comparison of Constrained Maximum Likelihood, Bayesian Estimation, and Fixed Reliability Approaches. *Psychological Methods*, *28*(3), 527–557. https://doi.org/10.1037/met0000435

Ulloa, J. G. (2018). *Applied Biomechatronics Using Mathematical Models*. Academic Press.

van de Schoot, R., Broere, J. J., Perryck, K. H., Zondervan-Zwijnenburg, M., & van Loey, N. E. (2015). Analyzing small data sets using Bayesian estimation: the case of posttraumatic stress symptoms following mechanical ventilation in burn survivors. *European Journal of Psychotraumatology*, *6*(1), 25216–25216. https://doi.org/10.3402/ejpt.v6.25216

van de Schoot, R., Depaoli, S., King, R., Kramer, B., Märtens, K., Tadesse, M. G., Vannucci, M., Gelman, A., Veen, D., Willemsen, J., & Yau, C. (2021). Bayesian statistics and

modelling. *Nature Reviews Methods Primers*, *1*(1). https://doi.org/10.1038/s43586-020-00001-2

van de Schoot, R., Kaplan, D., Denissen, J., Asendorpf, J. B., Neyer, F. J., & van Aken, M. A. G. (2014). A Gentle Introduction to Bayesian Analysis: Applications to Developmental Research. *Child Development*, *85*(3), 842–860. https://doi.org/10.1111/cdev.12169

van de Schoot, R., Depaoli, S., King, R., Kramer, B., Märtens, K., Tadesse, M. G., Vannucci, M., Gelman, A., Veen, D., Willemsen, J., & Yau, C. (2021). Bayesian statistics and modelling. *Nature Reviews Methods Primers*, *1*(1). https://doi.org/10.1038/s43586-020-00001-2

van Erp, S., & Browne, W. J. (2021). Bayesian Multilevel Structural Equation Modeling: An Investigation into Robust Prior Distributions for the Doubly Latent Categorical Model. *Structural Equation Modeling: A Multidisciplinary Journal*, *28*(6), 875-893. https://doi.org/10.1080/10705511.2021.1915146

van Erp, S., Mulder, J., & Oberski, D. L. (2018). Prior sensitivity analysis in default Bayesian structural equation modeling. *Psychological Methods*, *23*(2), 363–388. https://doi.org/10.1037/met0000162

van Ravenzwaaij, D., Cassey, P., & Brown, S. D. (2018). A simple introduction to Markov Chain Monte–Carlo sampling. *Psychonomic Bulletin & Review*, *25*(1), 143-154. https://doi.org/10.3758/s13423-016-1015-8

Vasishth, S., & Nicenboim, B. (2016). Statistical methods for linguistic research: Foundational ideas–Part I. *Language and Linguistics Compass*, *10*(8), 349-369. https://doi.org/10.1111/lnc3.12201

Wackerly, D. D., Mendenhall, W., & Scheaffer, R. L. (2008). *Mathematical statistics with applications* (Vol. 7). Thomson Brooks/Cole.

Wasserman, L. (2004). *Parametric Inference*. In: All of Statistics. Springer Texts in Statistics. Springer. https://doi.org/10.1007/978-0-387-21736-9_9

Wheadon, C. (2013). Using modern test theory to maintain standards in public qualifications in England. *Research Papers in Education*, *28*(5), 628–647. https://doi.org/10.1080/02671522.2012.706631

Winter, S. D., & Depaoli, S. (2022). Sensitivity of Bayesian model fit indices to the prior specification of latent growth models. *Structural Equation Modeling: A Multidisciplinary Journal*, *29*(5), 667-686, https://doi.org/10.1080/10705511.2022.2032078

Wilson, E. B., & Hilferty, M. M. (1931). The distribution of chi-square. *Proceedings of the National Academy of Sciences*, *17*(12), 684-688.

Wolfram MathWorld. (n.d.). *Regularized gamma function*. Retrieved from: https://mathworld.wolfram.com/RegularizedGammaFunction.html

Zheng, S., Zhang, L., Jiang, Z., & Pan, J. (2023). The influence of using inaccurate priors on Bayesian multilevel estimation. *Structural Equation Modeling: A Multidisciplinary Journal*, *30*(3), 429-448. https://doi.org/10.1080/10705511.2022.2136185

Table 1

Similarities and Differences between Frequentist and Bayesian Statistics

	Frequentist statistics	**Bayesian statistics**
p value	The probability that the null hypothesis is true in the population with the same or more extreme data are observed	The probability of the null hypothesis
Sample size	Large sample size is needed	Not necessary for large sample.
Prior information	Not included	Included
Population parameter	One true population parameter	A distribution of values reflecting uncertainty
Uncertainty	Uncertainty is defined by the sampling distribution based on infinite repeated sampling	Uncertainty is defined by the probability distribution for the population parameter
Estimated intervals	Confidence interval: 95% of the infinite samples taken from the population contain the true population value	Credible interval: A 95% probability of the population value within the interval
What does it ask	Is the hypothesis true or false?	Given the data, what is the probability of the assumption?
What does it give	*p*-value	The probability of supporting or opposing the assumption
Main advantages	1) Simple and easy to data analysis 2) Widely accepted in education and other behavioral and social science research	1) More intuitive results interpretation 2) Better equipped to model smaller sample sizes 3) Friendly to addressing model complexity (e.g., MLMs, SEMs) 4) Updating and accumulating knowledge
Main disadvantages	1) Sample sizes impact the significance/meaningful results; 2) Only offering yes/no answers	1) Different prior settings can yield varying results, so sensitivity analysis may be required 2) Advanced statistical packages and software require researchers to have more advanced statistical knowledge

Note. Adapted from van de Schoot, R., Kaplan, D., Denissen, J., Asendorpf, J. B., Neyer, F. J., & Van Aken, M. A. (2014). A gentle introduction to Bayesian analysis: Applications to developmental research. *Child Development*, *85*(3), 842-860.

Table 2

Overview of the Common Probability Distributions and Their Priors

Prior	Description	Conditions
Chi-square	- Continuous probability distributions - A special case of the gamma with shape parameter $\alpha = v/2$, and rate parameter $\beta = 1/2$	Often used for variance component
Gamma	- Right-skewed, shape parameter α (or a) and a rate parameter β (or b) - With higher α values, the distribution's average shifts right	- Conjugate prior of Poisson distribution - Used for variance component for Bayesian hierarchical models
Inverse Gamma	- Positive and right skewed with shape α (> 0) and scale parameter β (> 0) - More concentrated near zero - With higher α values, the distribution's average shifts left	- Mean known, variance unknown - Prior distribution of the mean is Gaussian - Conjugate prior for the variance parameter - Often used as an informative prior - Popular weakly informative *IG*(.01,.01)
Normal	- Have negative and positive values - Symmetric and centered at its location parameter μ with a spread governed by its scale parameter σ	- Mean unknown, variance known - Prior distribution of the mean is Gaussian - Conjugate prior - Used for coefficient parameters (i.e., intercepts and slopes)
Half-*t*	- Only positive half - Degree of freedom v, location μ, scale σ	Used for variance components
Half-Cauchy	- Right half of the half-*t* with v = 1 - Location μ and scale parameters σ - The tail is thick relative to other half-*t* distributions with v > 1	- Mean known, variance unknown - Prior distribution of the mean is Gaussian - Used for variance components for Bayesian hierarchical models
Uniform	- Rectangular distribution with parameters α or a (minimum) and β or b (maximum) - $X \sim U(\alpha, \beta)$	- Flat/uninformative prior with applications to the Gaussian data distribution - With a narrow uniform prior, the prior and posterior roughly match - With a wide uniform prior, the posterior matches the data distribution

Note. Adapted from "Common probability distributions and their priors," by D. Kaplan, 2023, in *Bayesian statistics for the social sciences* (pp. 31-46). Guilford Publications.

Table 3

Relationships among Families of Distributions Commonly used as Priors in Bayesian Analyses

Distribution	**Chi**	**G**	**IG**	**N**	**Half-*t***	**Half-C**	**U**
Chi-square (Chi)		✓	×	✓	×	×	✓
Gamma (G)	✓		✓	✓	×	×	✓
Inverse Gamma (IG)	×	✓		✓	×	×	✓
Normal (N)	✓	✓	✓		×	×	✓
Folded/Half-*t* (Half-*t*)	×	×	×	✓		✓	✓
Half-Cauchy (Half-C)	×	×	×	✓	✓		✓
Uniform (U)	✓	✓	✓	✓	✓	✓	

Note. A ✓ mark indicates direct transformability. The × mark indicates a relationship that can lead to equivalency but not transformability.

Table 4

Overview of Key Software Tools for Bayesian Statistical Analysis

Software	Description	MCMC Method	Key Features
ADMB	Originally designed for complex nonlinear statistical modeling of data, using automatic differentiation.	Suitable for a variety of statistical methods	Good for hierarchical and nonlinear statistical models, often used in ecology and resource management
BayesiaLab	An application for manipulating and visualizing Bayesian networks.	Various, including custom algorithms	Strong graphical interface, good for Bayesian networks and decision analysis
BUGS project (Bayesian inference Using Gibbs Sampling)	Initiated in 1989 by the MRC (Medical Research Council) Biostatistical Unit at the University of Cambridge.	Gibbs Sampler	Good for educational purposes and simpler Bayesian models.
INLA (Integrated nested Laplace approximation)	Software for Bayesian inference using the integrated nested Laplace approximations approach	Laplace Approximations	- Specializes in fast Bayesian inference for latent Gaussian models and hierarchical models - Ideal for scenarios where traditional MCMC methods are slow - Integrates with R via the R-INLA package
JAGS (Just Another Gibbs Sampler)	An open-source program similar to WinBUGS but more flexible and extensible	A combination of Metropolis sampling, Gibbs sampling, and other MCMC algorithms	Allows customization of models, integrates with R via the rjags package
MultiBUGS	A version of OpenBUGS designed to handle larger models by utilizing parallel processing	Gibbs Sampler	- Enables complex and large-scale - Bayesian model analysis by distributing computations across multiple processors
Mplus	Software for statistical modeling, commonly used in social sciences	Variety, depending on the model	- Focus on latent variable models - Supports diverse Bayesian analyses - User-friendly interface
OpenBUGS	Open-source version of WinBUGS	Gibbs Sampler	Extends WinBUGS functionality; integrates well with R; available in Windows, Mac OS, and Linux
PyMC	A Python library for probabilistic programming	Hamiltonian Monte Carlo (HMC), No-U-Turn Sampler (NUTS)	- Flexible modeling for defining custom probability distributions, hierarchical models, and non-standard models - Integrated well with NumPy, pandas, and SciPy - Includes visualization tools, like ArviZ for posterior distributions, trace plots, and diagnostic checks

Table 4 Continued

R	A free software environment for statistical computing and graphics	Varies by package	Highly extensible, integrates with multiple Bayesian tools like R2WinBUGS, R2OpenBUGS, and INLA
SAS/STAT	A component of the SAS software suite that includes procedures for Bayesian analysis	Gibbs Sampler，Metropolis-Hastings	Well-integrated into SAS's broader statistical analysis capabilities
Stan	An open-source C++ library with interfaces to R, Python, Julia, MATLAB, Stata, and other languages	HMC, NUTS	Flexible with non-conjugate priors, supports complex models, efficient sampling.
Turing	A Julia library for probabilistic programming and Bayesian inference	HMC, NUTS	Leverages Julia's performance, supports probabilistic programming
WinBUGS	Early Bayesian analysis software using the MCMC	Gibbs Sampler	User-friendly and notable as an original desktop application for Windows

Table 5

Supported Bayesian Multilevel Models by R Packages

Packages	Continuous	Binary	Free	Available in R	Easy use (formula similar to 'lmer4')	Variance components priors included
arm	✓	✓	✓	✓	✓	✗
Bambi	✓	✓	✓	✗	✗	✗
bayesm	✓	✓	✓	✓	✓	✗
bayesmh	✓	✓	✓	✗ (Stata)	✗	✗
BayesReg	✓	✓	✓	✓	✓	✗
blme	✓	✓	✓	✓	✓	✓
brms	✓	✓	✓	✓	✓	✓
BRugs	✓	✓	✓	✓	✗	✓
DPpackage	✓	✓	✓	✗ (Discontinued)	✗	✗
glmmADMB	✓	✓	✓	✓	✓	✗
glmmAK	✗	✓	✓	✗ (Discontinued)	✗	✗
glmmBUGS	✓	✓	✓	✗ (Discontinued)	✗	✗
LaplacesDemon	✓	✓	✓	✓	✗	✓
MCMCglmm	✓	✓	✓	✓	✓	✓
MCMCpack	✓	✓	✓	✓	✗	✓
Mplus Bayes	✓	✓	✗	✗ (Mplus)	✗	✓
R2admb	✓	✓	✓	✓	✗	✗
R2BayesX	✓	✗	✓	✓	✗	✗
R-INLA	✓	✓	✓	✓	✓	✓
rjags	✓	✓	✓	✓	✗	✓
R2MLwiN	✓	✓	✓	✗ (MLwiN)	✗	✗
R2OpenBUGS	✓	✓	✓	✓ (OpenBUGS)	✗	✓
R2WinBUGS	✓	✓	✓	✓ (WinBUGS)	✗	✓
rstan	✓	✓	✓	✓	✗	✓
rstanarm	✓	✓	✓	✓	✓	✓

Note. Continuous and binary indicate whether the certain package can handle continuous and binary data.

Table 6

Overview of Previous Research on Bayesian Variance Priors

Prior	Main Findings	Studies
Gamma	Provides better control over variance estimates, reducing the bias especially in settings with moderate to large sample sizes	Baldwin & Fellingham (2013); Gustafson et al. (2006); Lee et al. (2020)
Inverse gamma	- *IG*(0.01,0.01) and half- Cauchy (0,4) had good posteriors - Informative IG outperformed frequentist approach - Weak informative priors yielded biased variance estimates	Bolin et al. (2019); Depaoli & Clifton (2015); Gelman (2006); McNeish (2016)
Normal	- Not commonly used for variance components due to their symmetrical nature around zero - Be effective in structural equation modeling for smaller loadings	Liang (2020)
Half-*t*	Provides robustness against outliers and heavy tails, preventing overestimation of variance components in small samples	Bürkner (2017a,b)
Half-Cauchy	Half-Cauchy may be better than IG as a variance prior for parameter values closer to zero	Gelman (2006); McNeish & Stapleton (2016)
Uniform	- Typically performs well for J clusters ≥ 3 for proper posterior density - With large *sd* is recommended - May overestimate variance components with few clusters	Baldwin & Fellingham (2013); Danisels (1999); Gelman (2006); McNeish & Stapleton (2016)

Table 7

False Positive Rate and Correct Positive for L1 Predictor Slope Coefficient, Collapsed across L2 Predictor Effects

Condition	False Positive Rate							Correct Positive (L1 Coeff *R*-squared = .10)							Correct Positive (L1 Coeff *R*-squared = .25)						
	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab
ICC = .01																					
J = 10																					
M = 5	.07	.06	.04	.04	.05	.04	.05	.67	.88	.86	.86	.87	.86	.87	.99	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.02	.03	.03	.03	.03	.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 30																					
M = 5	.07	.04	.04	.04	.04	.04	.04	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.05	.06	.06	.06	.06	.06	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 100																					
M = 5	.03	.04	.04	.04	.03	.04	.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.05	.05	.05	.05	.05	.05	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ICC = .20																					
J = 10																					
M = 5	.06	.06	.04	.04	.03	.04	.03	.63	.85	.81	.81	.77	.81	.77	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.03	.03	.03	.02	.03	.02	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 30																					
M = 5	.07	.04	.04	.04	.06	.04	.06	.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.06	.10	.10	.10	.11	.10	.11	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 100																					
M = 5	.05	.04	.04	.04	.04	.04	.04	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.04	.05	.05	.05	.05	.05	.05	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ICC = .40																					
J = 10																					
M = 5	.06	.06	.06	.06	.03	.06	.03	.62	.85	.78	.78	.75	.78	.75	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.03	.02	.02	.02	.02	.02	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 30																					
M = 5	.07	.07	.07	.07	.08	.07	.08	.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.06	.11	.11	.11	.11	.11	.11	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
J = 100																					
M = 5	.05	.03	.03	.03	.03	.03	.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.04	.05	.05	.05	.05	.05	.05	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
***J* = 10**	**.06**	**.04**	**.04**	**.04**	**.03**	**.04**	**.03**	**.82**	**.93**	**.91**	**.91**	**.90**	**.91**	**.90**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
***J* = 100**	**.04**	**.04**	**.04**	**.04**	**.04**	**.04**	**.04**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Total	**.05**	**.05**	**.05**	**.05**	**.05**	**.05**	**.05**	**.94**	**.98**	**.97**	**.97**	**.97**	**.97**	**.97**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

Note. Cell means above based on *N* = 600 replications (200 replications per condition x three levels of L2 Total *R*-squared values). Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = IG(.01, .01) used for variance component priors, VARIGab = IG(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Table 8

False Positive Rates for L2 Intercept Coefficient and Raw Bias for L2 Intercept and Variance

Condition	False Positive Rate							L2 Intercept Coeff Raw Bias							L2 Intercept Variance Raw Bias						
	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab
ICC = .01																					
J = 10																					
M = 5	.00	.00	.03	.03	.03	.03	.03	.01	.01	.04	.04	.02	.04	.02	.00	.13	.03	.03	.01	.03	.01
M = 30	.01	.01	.02	.02	.03	.02	.03	.00	.00	.02	.02	.02	.02	.02	.00	.02	.01	.01	.00	.01	.00
J = 30																					
M = 5	.00	.01	.02	.02	.04	.02	.04	.00	-.02	-.02	-.02	-.02	-.02	-.02	.00	.04	.02	.02	.01	.02	.01
M = 30	.02	.03	.04	.04	.04	.04	.04	.00	.00	-.01	-.01	-.01	-.01	-.01	.00	.01	.01	.01	.00	.01	.00
J = 100																					
M = 5	.02	.04	.04	.04	.04	.04	.04	.00	.00	.00	.00	.00	.00	.00	.01	.02	.01	.01	.01	.01	.01
M = 30	.05	.04	.04	.04	.04	.04	.04	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
ICC = .20																					
J = 10																					
M = 5	.04	.00	.05	.05	.01	.05	.01	.04	.02	.03	.03	.07	.03	.07	-.13	.21	-.06	-.06	.07	-.06	.07
M = 30	.13	.01	.06	.06	.03	.06	.03	-.01	.01	.03	.03	.03	.03	.03	-.08	.16	.01	.01	.06	.01	.06
J = 30																					
M = 5	.06	.04	.05	.05	.04	.05	.04	-.01	-.02	-.02	-.02	-.02	-.02	-.02	-.07	.01	-.05	-.05	.02	-.05	.02
M = 30	.08	.03	.04	.04	.03	.04	.03	.00	.02	.01	.01	.02	.01	.02	-.03	.02	.00	.00	.01	.00	.01
J = 100																					
M = 5	.07	.05	.05	.05	.05	.05	.05	.00	.00	.00	.00	.00	.00	.00	-.02	.00	-.01	-.01	.01	-.01	.01
M = 30	.06	.07	.07	.07	.07	.07	.07	.00	.00	.00	.00	.00	.00	.00	-.01	.00	-.01	-.01	.00	-.01	.00
ICC = .40																					
J = 10																					
M = 5	.10	.01	.06	.06	.02	.06	.02	.05	.02	.03	.03	.07	.03	.07	-.28	.41	-.10	-.10	.14	-.10	.14
M = 30	.14	.01	.06	.06	.03	.06	.03	-.01	.02	.03	.03	.04	.03	.04	-.17	.40	.05	.05	.16	.05	.16
J = 30																					
M = 5	.07	.05	.06	.06	.05	.06	.05	-.02	-.02	-.02	-.02	-.03	-.02	-.03	-.10	.03	-.05	-.05	.04	-.05	.04
M = 30	.06	.03	.04	.04	.03	.04	.03	.00	.02	.02	.02	.02	.02	.02	-.06	.04	.00	.00	.04	.00	.04
J = 100																					
M = 5	.08	.07	.07	.07	.07	.07	.07	.00	.00	.00	.00	.00	.00	.00	-.02	.00	-.02	-.02	.00	-.02	.00
M = 30	.06	.07	.08	.08	.07	.08	.07	.01	.00	.00	.00	.00	.00	.00	-.02	.00	-.01	-.01	.00	-.01	.00
***J* = 10**	**.07**	**.01**	**.05**	**.05**	**.03**	**.05**	**.03**	**.01**	**.01**	**.03**	**.03**	**.04**	**.03**	**.04**	**-.11**	**.22**	**-.01**	**-.01**	**.07**	**-.01**	**.07**
***J* = 100**	**.05**	**.06**	**.06**	**.06**	**.06**	**.06**	**.06**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**.00**	**-.01**	**.00**	**-.01**	**-.01**	**.00**	**-.01**	**.00**
Total	**.06**	**.03**	**.05**	**.05**	**.04**	**.05**	**.04**	**.00**	**.00**	**.01**	**.01**	**.01**	**.01**	**.01**	**-.05**	**.08**	**-.01**	**-.01**	**.03**	**-.01**	**.03**

Note. Cell means above based on *N* = 1,800 replications (200 replications per condition x nine crossed levels of L1 and L2 Total *R*-squared values). Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = IG(.01, .01) used for variance component priors, VARIGab = IG(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Table 9

False Positive Rate and Correct Positive for L2 Predictor Slope Coefficient, Collapsed across L1 Predictor Effects

Condition	False Positive Rate							Correct Positive (L2 Coeff Rsq = .10)							Correct Positive (L2 Coeff Rsq = .25)						
	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab	ML	BUI	BI-N	BI-G01	BI-Gab	BI-N+G01	BI-N+Gab
ICC = .01																					
J = 10																					
M = 5	.00	.01	.07	.07	.06	.07	.06	.00	.02	.08	.08	.07	.08	.07	.01	.02	.09	.09	.07	.09	.07
M = 30	.02	.01	.04	.04	.05	.04	.05	.03	.03	.07	.07	.10	.07	.10	.03	.05	.09	.09	.18	.09	.18
J = 30																					
M = 5	.01	.09	.10	.10	.10	.10	.10	.02	.11	.12	.12	.12	.12	.12	.02	.12	.13	.13	.15	.13	.15
M = 30	.04	.06	.07	.07	.08	.07	.08	.07	.17	.19	.19	.20	.19	.20	.09	.27	.28	.28	.33	.28	.33
J = 100																					
M = 5	.03	.16	.17	.17	.16	.17	.16	.03	.21	.21	.21	.21	.21	.21	.04	.23	.23	.23	.25	.23	.25
M = 30	.03	.09	.08	.08	.08	.08	.08	.12	.35	.33	.33	.33	.33	.33	.21	.60	.58	.58	.61	.58	.61
ICC = .20																					
J = 10																					
M = 5	.04	.01	.06	.06	.02	.06	.02	.07	.05	.18	.18	.07	.18	.07	.12	.14	.34	.34	.24	.34	.24
M = 30	.16	.01	.07	.07	.01	.07	.01	.27	.06	.17	.17	.09	.17	.09	.51	.29	.53	.53	.42	.53	.42
J = 30																					
M = 5	.09	.08	.12	.12	.08	.12	.08	.18	.35	.38	.38	.33	.38	.33	.37	.68	.69	.69	.68	.69	.68
M = 30	.07	.06	.08	.08	.06	.08	.06	.43	.64	.67	.67	.65	.67	.65	.88	.97	.98	.98	.98	.98	.98
J = 100																					
M = 5	.10	.06	.07	.07	.06	.07	.06	.44	.67	.67	.67	.67	.67	.67	.84	.99	.99	.99	.99	.99	.99
M = 30	.04	.06	.07	.07	.06	.07	.06	.90	.99	.99	.99	.99	.99	.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ICC = .40																					
J = 10																					
M = 5	.10	.03	.08	.08	.02	.08	.02	.18	.06	.22	.22	.07	.22	.07	.31	.24	.49	.49	.38	.49	.38
M = 30	.13	.03	.05	.05	.02	.05	.02	.30	.07	.13	.13	.07	.13	.07	.57	.30	.53	.53	.42	.53	.42
J = 30																					
M = 5	.10	.06	.08	.08	.06	.08	.06	.31	.55	.57	.57	.53	.57	.53	.67	.91	.93	.93	.91	.93	.91
M = 30	.07	.07	.09	.09	.07	.09	.07	.49	.73	.76	.76	.75	.76	.75	.92	.99	1.00	1.00	.99	1.00	.99
J = 100																					
M = 5	.09	.07	.08	.08	.07	.08	.07	.72	.93	.94	.94	.93	.94	.93	.99	1.00	1.00	1.00	1.00	1.00	1.00
M = 30	.05	.07	.07	.07	.07	.07	.07	.93	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
***J* = 10**	**.07**	**.02**	**.06**	**.06**	**.03**	**.06**	**.03**	**.14**	**.05**	**.14**	**.14**	**.08**	**.14**	**.08**	**.26**	**.17**	**.34**	**.34**	**.29**	**.34**	**.29**
***J* = 100**	**.06**	**.08**	**.09**	**.09**	**.08**	**.09**	**.08**	**.52**	**.69**	**.69**	**.69**	**.69**	**.69**	**.69**	**.68**	**.80**	**.80**	**.80**	**.81**	**.80**	**.81**
Total	**.06**	**.06**	**.08**	**.08**	**.06**	**.08**	**.06**	**.31**	**.39**	**.43**	**.43**	**.40**	**.43**	**.40**	**.48**	**.54**	**.60**	**.60**	**.59**	**.60**	**.59**

Note. Cell means above based on *N* = 600 replications (200 replications per condition x three levels of L1 Total *R*-squared values).). Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = IG(.01, .01) used for variance component priors, VARIGab = IG(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Table 10

Credible Interval Coverage Results for Variance Components for Conditions with J = 30

Model-Estimated Result	*M* = 5		*M* = 30		Average	
	BUI	BI-Gab	BUI	BI-Gab	BUI	BI-Gab
L1 Resid Var 95% CI Coverage	0.95	0.94	0.94	0.94	0.95	0.94
L2 Resid Var 95% CI Coverage	0.96	0.93	0.97	0.92	0.96	0.93
L1 Resid Var 95% CI Width	0.22	0.22	0.08	0.08	0.15	0.15
L2 Resid Var 95% CI Width	0.48	0.39	0.30	0.26	0.39	0.32

Note. Results for replications involving conditions with *J* = 30 (*N* = 200 replications per cell) from R brms package using median point estimates and equal-tailed posterior cutoffs.

Table 11

Estimation Time for the Bayesian Analyses Focusing on Inverse Gamma Prior Informativeness for TIMSS Data

	Approach	Full sample	Subsample
1	BUI	10.10 min	28.24 sec
2	BI Coef N	11.91 min	28.06 sec
3	BI VarIG01	13.21 min	27.32 sec
4	BI VarIGab	11.39 min	25.05 sec
5	BI Coef + VarIG01	8.84 min	27.20 sec
6	BI Coef + VarIGab	9.99 min	25.24 sec

Note. min = minutes; sec = seconds.

Table 12

Steps for Calculating Inverse Gamma Prior Informativeness using Full Sample for Bayesian Multilevel Analysis

Step/Parameter	Known Distribution Properties	L2 Intercept Variance Example	L1 Residual Variance Example
Chi-square			
Mean (df)	k	273 ($n-1$)	8423 ($n-1$)
Mode	$max(k-2,0)$	$max(271,0) = 271$	$max(8421,0) = 8421$
Variance	$2k$	$2 \times 273 = 546$	$2 \times 8423 = 16846$
Scaling Factor (W)	$(s^2 + 2)/k$	$(12.38^2 + 2) / 273 = 0.5708$	$(46.19^2 + 2) / 8423 = 0.2536$
Scaled Chi-Square			
Scaled Mean (Wk)	$W * k$	$0.5708 \times 272 = 155.2644$	$0.2536 \times 8422 = 2135.516$
Scaled Mode ($Wk-2$)	$max(W*k-2,0)$	$max(153.2644,0)$	max (2133.516, 0)
Scaled Variance ($2Wk$)	$2W*k$	$2 \times 155.2644 = 310.5288$	$2 \times 2135.516 = 4271.032$
Chi-square Prior		In Var for Mplus Chi(155) In SDs for brms Chi(12)	In Var for Mplus Chi(2136) In SDs for brms Chi(46)
Gamma			
Alpha (a)	$W*k/2$	$272 \times 0.5708 / 2 = 77.6322$	$8422 \times 0.2536 / 2 = 1067.758$
Theta	2	2	2
Beta (b, $1/theta$)	0.5	0.5	0.5
Gamma Mean	$a*theta$	$77.6322 \times 2 = 155.2644$	$1067.758 \times 2 = 2135.516$
Gamma Mode	$max((a-1) \times theta, 0)$	$max((77.6322-1) \times 2), 0)$	$max((1066.758 \times 2), 0)$
Gamma Variance	$a*\theta^2$	$77.6322 \times (2^2) = 310.5288$	$1067.758 \times (2^2) = 4271.032$
Gamma Prior		In Var for Mplus *G*(78, 0.5) In SDs for brms *G*(9, 0.5)	In Var for Mplus *G*(1068, 0.5) In SDs for brms *G*(33, 0.5)
Inverse gamma			
IG Alpha	$(mean+)/2$	$(155.2644 + 153.2644)/2 = 154.26$	$(2135.516 + 2133.516)/2 = 2134.52$
IG Beta	$mean*(a-1)$	$155.2644 \times (154.26-1) = 23796.51$	$2135.516 \times (2134.52-1) = 4556158$
IG Mean	$b/(a-1)$	$23796.51 / (154.26-1) = 155.2644$	$4556158 / (2134.52-1) = 2135.516$
IG Mode	$b/(a+1)$	$23796.51 / (154.26+1) = 153.2644$	$4556158 / (2134.52+1) = 2133.516$
IG Variance	$b^2/((a^2\text{-}1)*(a\text{-}2))$	$\frac{23796.51^2}{((154.26^2-1) \times (154.26-2))} = 158.32$	$\frac{4556158^2}{((2134.52^2-1) \times (2134.52-2))} = 2138.52$
IG Prior		In Var for Mplus *IG*(154, 23797) In SDs for brms *IG*(12, 154)	In Var for Mplus *IG*(2134, 4552216) In SDs for brms *IG*(46, 2135)

Table 13

Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Inverse Gamma Prior Informativeness (Full Sample)

	BUI					BI Coef N					BI VarIG01				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	523.22	(0.88)	521.45	525.00	3.55	523.21	(0.90)	521.48	524.95	3.47	523.22	(0.94)	521.38	525.00	3.62
L1 Math (*Z*)	59.13	(0.52)	58.12	60.14	2.02	59.10	(0.51)	58.15	60.07	1.92	59.13	(0.50)	58.14	60.11	1.97
L2 Math (*Z*)	64.20	(0.95)	62.29	66.12	3.83	64.19	(0.95)	62.30	66.03	3.73	64.19	(0.96)	62.29	66.04	3.75
L1 x L2 Interact	-2.72	(0.56)	-3.82	-1.62	2.20	-2.74	(0.58)	-3.82	-1.65	2.17	-2.74	(0.55)	-3.85	-1.61	2.24
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	12.38	(0.82)	10.87	14.04	3.17	12.40	(0.79)	10.90	14.04	3.14	12.37	(0.81)	10.88	13.98	3.10
Residual (L1)	46.20	(0.36)	45.53	46.89	1.36	46.19	(0.37)	45.50	46.92	1.42	46.19	(0.34)	45.51	46.90	1.39
	BI VarIGab					BI Coef + VarIG01					BI Coef + VarIGab				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	523.19	(0.90)	521.44	525.02	3.58	523.15	(0.92)	521.39	524.99	3.60	523.21	(0.89)	521.49	525.09	3.60
L1 Math (*Z*)	59.12	(0.47)	58.16	60.05	1.89	59.10	(0.50)	58.17	60.06	1.89	59.12	(0.49)	58.14	60.08	1.94
L2 Math (*Z*)	64.17	(0.97)	62.28	66.05	3.77	64.19	(0.97)	62.35	66.00	3.65	64.19	(0.95)	62.28	66.04	3.76
L1 x L2 Interact	-2.72	(0.54)	-3.80	-1.62	2.18	-2.74	(0.54)	-3.78	-1.65	2.13	-2.71	(0.54)	-3.83	-1.68	2.15
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	12.35	(0.76)	10.96	13.99	3.03	12.32	(0.81)	10.87	14.07	3.20	12.37	(0.74)	10.91	14.02	3.11
Residual (L1)	46.19	(0.36)	45.50	46.88	1.38	46.18	(0.35)	45.51	46.91	1.40	46.19	(0.34)	45.54	46.89	1.35

Note. Analysis involved *N* = 8698 students (Level 1, L1) within 273 schools (Level 2, L2). Any acronyms for outcomes/predictors here. All lower-level (L1) predictors were cluster-mean centered, and continuous predictors were standardized. Models were estimated using Bayesian Markov Chain Monte Carlo (MCMC) via the brms package in *R*. BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Table 14

Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Inverse Gamma Prior Informativeness (Subsample)

	BUI					BI Coef N					BI VarIG01				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	519.22	(3.87)	511.35	527.07	15.72	519.28	(3.87)	511.51	527.09	15.58	519.16	(3.66)	511.84	526.62	14.78
L1 Math (*Z*)	64.71	(3.68)	57.59	72.18	14.59	64.68	(3.72)	57.54	72.01	14.47	64.64	(3.64)	57.52	72.09	14.57
L2 Math (*Z*)	72.37	(3.94)	64.14	80.40	16.26	72.28	(3.92)	64.26	80.05	15.79	72.42	(3.82)	65.05	79.79	14.74
L1 x L2 Interact	-7.80	(4.22)	-16.21	0.29	16.50	-7.90	(4.18)	-15.76	0.24	16.00	-7.83	(4.23)	-16.23	0.86	17.09
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	5.75	(4.81)	0.29	17.19	16.90	5.46	(4.54)	0.31	16.14	15.83	0.38	(0.53)	0.01	10.40	10.39
Residual (L1)	45.22	(2.89)	40.29	50.99	10.70	45.24	(2.70)	40.39	51.06	10.67	45.28	(2.55)	40.37	51.11	10.74

	BI VarIGab					BI Coef + VarIG01					BI Coef + VarIGab				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	519.17	(3.83)	511.53	526.95	15.42	519.31	(3.75)	512.01	526.36	14.35	519.18	(3.89)	511.58	526.81	15.23
L1 Math (*Z*)	64.76	(3.74)	57.40	72.10	14.70	64.63	(3.79)	57.09	72.40	15.31	64.72	(3.83)	57.50	71.94	14.44
L2 Math (*Z*)	72.40	(4.07)	64.51	80.49	15.98	72.49	(3.76)	65.03	79.70	14.67	72.32	(4.08)	63.95	80.34	16.39
L1 x L2 Interact	-7.79	(4.32)	-16.29	0.38	16.67	-7.82	(4.19)	-15.02	0.04	15.06	-7.83	(4.20)	-15.90	0.13	16.03
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	5.64	(1.99)	2.94	11.40	8.46	0.44	(0.62)	0.01	9.85	9.84	5.57	(2.01)	2.80	11.77	8.97
Residual (L1)	45.07	(2.43)	40.65	50.39	9.74	45.28	(2.83)	40.58	50.91	10.33	45.12	(2.47)	40.74	50.11	9.37

Note. Analysis involved $N = 150$ students (Level 1, L1) within 30 schools (Level 2, L2). Any acronyms for outcomes/predictors here. All lower-level (L1) predictors were cluster-mean centered, and continuous predictors were standardized. Models were estimated using Bayesian Markov Chain Monte Carlo (MCMC) via the brms package in *R.* BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Table 15

Estimation Time for the Bayesian Analyses Focusing on G, IG, Half-t, and Half-Cauchy Informative Priors for TIMSS Data

	Approach	Full sample	Subsample
1	BUI	10.09 min	27.16 sec
2	BI VarGab	13.09 min	24.43 sec
3	BI VarIGab	7.56 min	24.74 sec
4	BI VarHalf-*t* *df*=1	8.59 min	27.16 sec
5	BI VarHalf-*t*	10.62 min	25.46 sec
6	BI VarHalf-Cauchy	10.93 min	26.56 sec

Note. min = minutes; sec = seconds.

Table 16

Steps for Calculating Strongly Informative Half-t and Half-Cauchy Priors using Full Sample for Bayesian Multilevel Analysis

Step/Parameter	Known Distribution Properties	L2 Intercept Variance Example	L1 Residual Variance Example
Chi-square			
Mean (df)	k	273 ($n-1$)	8423 ($n-1$)
Mode	$max(k-2,0)$	$max(271,0) = 271$	$max(8421,0) = 8421$
Variance	$2k$	$2 \times 273 = 546$	$2 \times 8423 = 16846$
Scaling Factor (W)	$(s^2 + 2)/k$	$(12.38^2 + 2) / 273 = 0.5708$	$(46.19^2 + 2) / 8423 = 0.2536$
Scaled Chi-Square			
Scaled Mean (Wk)	$W * k$	$0.5708 \times 272 = 155.2644$	$0.2536 \times 8422 = 2135.516$
Scaled Mode ($Wk-2$)	$max(W*k-2,0)$	$max(153.2644,0)$	max (2133.516, 0)
Scaled Variance ($2Wk$)	$2W*k$	$2 \times 155.2644 = 310.5288$	$2 \times 2135.516 = 4271.032$
Chi-square Prior		In Var for Mplus Chi(155) In SDs for brms Chi(12)	In Var for Mplus Chi(2136) In SDs for brms Chi(46)
Gamma			
Alpha (a)	$W*k/2$	$272 \times 0.5708 / 2 = 77.6322$	$8422 \times 0.2536 / 2 = 1067.758$
Theta	2	2	2
Beta (b, $1/theta$)	0.5	0.5	0.5
Gamma Mean	$a*theta$	$77.6322 \times 2 = 155.2644$	$1067.758 \times 2 = 2135.516$
Gamma Mode	$max((a-1) \times theta, 0)$	$max((77.6322-1) \times 2), 0)$	$max((1066.758 \times 2), 0)$
Gamma Variance	$a*\theta^2$	$77.6322 \times (2^2) = 310.5288$	$1067.758 \times (2^2) = 4271.032$
Gamma Prior		In Var for Mplus *G*(78,0.5) In SDs for brms *G*(9,0.5)	In Var for Mplus *G*(1068,0.5) In SDs for brms *G*(33,0.5)
Inverse gamma			
IG Alpha	$(mean+)/2$	$(155.2644 + 153.2644)/2 = 154.26$	$(2135.516 + 2133.516)/2 = 2134.52$
IG Beta	$mean*(a-1)$	$155.2644 \times (154.26-1) = 23796.51$	$2135.516 \times (2134.52-1) = 4556158$
IG Mean	$b/(a-1)$	$23796.51 / (154.26-1) = 155.2644$	$4556158 / (2134.52-1) = 2135.516$
IG Mode	$b/(a+1)$	$23796.51 / (154.26+1) = 153.2644$	$4556158 / (2134.52+1) = 2133.516$
IG Variance	$b^2/((a^2\text{-}1)*(a\text{-}2))$	$\frac{23796.51^2}{((154.26^2-1) \times (154.26-2))} = 158.32$	$\frac{4556158^2}{((2134.52^2-1) \times (2134.52-2))} = 2138.52$
IG Prior		In Var for Mplus *IG*(154,23797) In SDs for brms *IG*(12,154)	In Var for Mplus *IG*(2134,4552216) In SDs for brms *IG*(46,2135)

Table 16 Continued

Half-*t*			
Variance	s^2	$12.38^2 = 153.2644$	$46.19^2 = 2133.516$
Mean (df)	k	272	8422
Scaling Factor (W)	$(s^2 + 2)/k$	$(153.2644 + 2)/272 = 0.5708$	$(2133.516 + 2)/8422 = 0.2536$
SD	$\sqrt{2 * k * W}$	$\sqrt{2 \times 272 \times 0.5708} = 17.6218$	$\sqrt{2 \times 8422 \times 0.2536} = 65.3531$
Location	$\sqrt{Observed\ variance}$	$\sqrt{153.2644} = 12.38$	$\sqrt{2133.516} = 46.19$
Scale	$Standard\ deviation$	17.6218	65.3531
Half-*t* Prior	(*df*, location, scale)	In Var for Mplus half-*t*(1,153,311) In SDs for brms half-*t*(1,12,18), where *df* = 1 as an example	In Var for Mplus half-*t*(1,2134,4271) In SDs for brms half-*t*(1,46,65), where *df* = 1 as an example
Half-Cauchy	A special case of the half-*t* with *df* = 1 and a location parameter of 0 (0, scale)	In Var for Mplus half-Cauchy(0,311) In SDs for brms half-Cauchy(0,18)	In Var for Mplus half-Cauchy(0,4271) In SDs for brms half-Cauchy(0,65)

Table 17

Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Informative G, IG, Half-t, and Half-Cauchy (Full Sample)

	BUI					BI VarGab					BI VarIGab				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	523.22	(0.88)	521.45	525.00	3.55	523.23	(0.90)	521.44	525.01	3.57	523.22	(0.87)	521.53	525.02	3.49
L1 Math (*Z*)	59.13	(0.52)	58.12	60.14	2.02	59.14	(0.52)	58.16	60.08	1.92	59.12	(0.51)	58.13	60.11	1.98
L2 Math (*Z*)	64.20	(0.95)	62.29	66.12	3.83	64.15	(0.94)	62.21	66.03	3.82	64.13	(0.97)	62.19	66.05	3.86
L1 x L2 Interact	-2.72	(0.56)	-3.82	-1.62	2.20	-2.71	(0.54)	-3.78	-1.64	2.14	-2.72	(0.55)	-3.82	-1.67	2.15
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	12.38	(0.82)	10.87	14.04	3.17	12.47	(0.79)	10.96	14.15	3.19	12.40	(0.77)	11.10	14.06	2.96
Residual (L1)	46.20	(0.36)	45.53	46.89	1.36	46.22	(0.34)	45.54	46.88	1.34	46.18	(0.34)	45.50	46.88	1.38
	BI VarHalf-*t* *df*=1					BI VarHalf-*t*					BI VarHalf-Cauchy				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	523.22	(0.91)	521.42	525.04	3.62	523.19	(0.88)	521.50	524.92	3.42	523.25	(0.88)	521.43	524.98	3.55
L1 Math (*Z*)	59.13	(0.49)	58.17	60.05	1.88	59.12	(0.49)	58.12	60.10	1.98	59.12	(0.48)	58.15	60.06	1.91
L2 Math (*Z*)	64.19	(0.96)	62.27	66.00	3.73	64.14	(0.95)	62.30	65.98	3.68	64.16	(0.95)	62.20	66.00	3.80
L1 x L2 Interact	-2.72	(0.55)	-3.85	-1.59	2.26	-2.73	(0.55)	-3.81	-1.61	2.20	-2.73	(0.55)	-3.78	-1.67	2.11
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	12.39	(0.80)	10.90	14.06	3.16	12.41	(0.83)	10.93	14.09	3.16	12.40	(0.80)	10.94	14.08	3.14
Residual (L1)	46.19	(0.36)	45.50	46.91	1.41	46.18	(0.36)	45.50	46.91	1.41	46.19	(0.36)	45.51	46.92	1.41

Note. Analysis involved N = 8698 students (Level 1, L1) within 273 schools (Level 2, L2). Any acronyms for outcomes/predictors here. All lower-level (L1) predictors were cluster-mean centered, and continuous predictors were standardized. Models were estimated using Bayesian Markov Chain Monte Carlo (MCMC) via the brms package in R. BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution G(a,b) priors for variance components; parameters a and b derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma IG(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-t(df, location, scale) used for variance component priors, with degree of freedom df, location, and scale parameters based on chi-square distribution (df = 1 or df = k). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-t distribution with one degree of freedom and a location parameter of zero.

Table 18

Parameter Estimates for 2-Level Bayesian Multilevel Modeling Using Informative G, IG, Half-t, and Half-Cauchy (Subsample)

	BUI					BI VarGab					BI VarIGab				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	519.22	(3.87)	511.35	527.07	15.72	519.26	(4.10)	511.37	527.28	15.91	519.13	(3.96)	511.19	526.84	15.65
L1 Math (*Z*)	64.71	(3.68)	57.59	72.18	14.59	64.65	(3.84)	57.19	72.44	15.25	64.71	(3.80)	57.38	72.05	14.67
L2 Math (*Z*)	72.37	(3.94)	64.14	80.40	16.26	72.33	(4.04)	64.11	80.31	16.20	72.36	(4.08)	64.56	80.16	15.60
L1 x L2 Interact	-7.80	(4.22)	-16.21	0.29	16.50	-7.87	(4.23)	-16.31	0.63	16.94	-7.79	(4.24)	-15.84	0.36	16.20
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	5.75	(4.81)	0.29	17.19	16.90	6.30	(2.87)	2.11	13.03	10.92	5.64	(2.02)	2.86	11.58	8.72
Residual (L1)	45.22	(2.89)	40.29	50.99	10.70	46.47	(2.78)	41.32	52.76	11.44	45.10	(2.51)	40.66	50.43	9.77
	BI VarHalf-*t* *df*=1					BI VarHalf-*t*					BI VarHalf-Cauchy				
Fixed Effects	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*	*Est*	*(SE)*	*LB*	*UB*	*Width*
Intercept (Mean)	519.18	(3.85)	511.50	527.05	15.55	519.25	(3.96)	511.16	526.86	15.70	519.20	(3.86)	511.75	526.56	14.81
L1 Math (*Z*)	64.67	(3.93)	57.17	72.38	15.21	64.56	(3.67)	57.29	71.94	14.65	64.68	(3.69)	57.58	71.95	14.37
L2 Math (*Z*)	72.31	(4.06)	64.23	80.29	16.06	72.39	(4.01)	64.47	79.99	15.52	72.40	(3.89)	64.71	80.25	15.54
L1 x L2 Interact	-7.73	(4.15)	-15.95	0.60	16.55	-7.83	(4.19)	-16.32	0.77	17.09	-7.74	(4.22)	-16.07	0.18	16.25
Random Effects	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*	*SD*	*(SE)*	*LB*	*UB*	*Width*
Intercept (L2)	5.33	(3.83)	0.47	14.19	13.72	5.40	(4.08)	0.32	14.96	14.64	4.81	(4.04)	0.23	15.31	15.08
Residual (L1)	45.29	(2.72)	40.41	51.00	10.59	45.23	(2.65)	40.29	50.96	10.67	45.10	(2.66)	40.26	50.90	10.64

Note. Analysis involved *N* = 8698 students (Level 1, L1) within 273 schools (Level 2, L2). Any acronyms for outcomes/predictors here. All lower-level (L1) predictors were cluster-mean centered, and continuous predictors were standardized. Models were estimated using Bayesian Markov Chain Monte Carlo (MCMC) via the brms package in R. BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and b derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = k). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-t distribution with one degree of freedom and a location parameter of zero.

Figure 1

Comparison of Regression Parameter Estimates with Frequentist and Bayesian Approaches

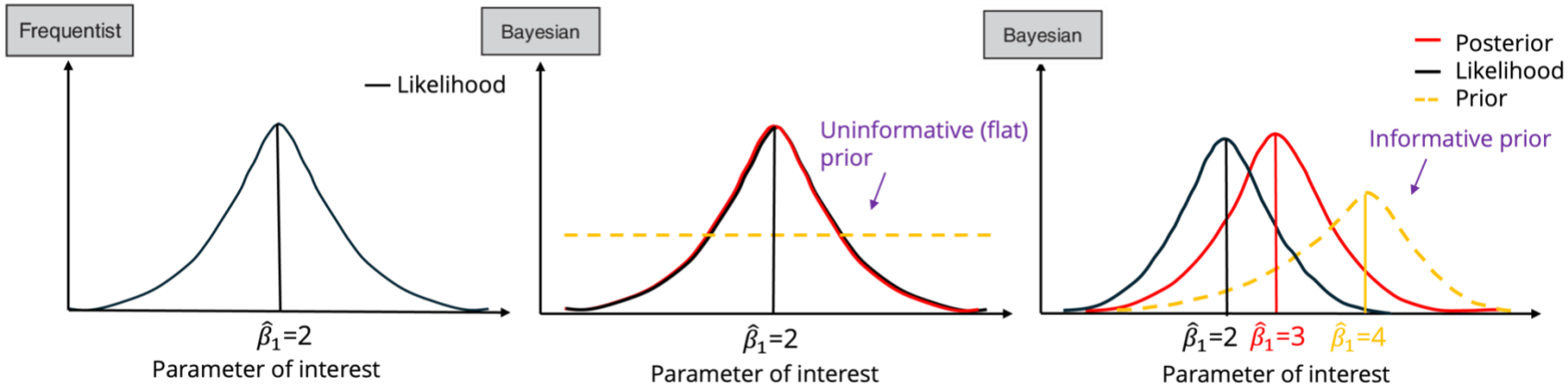


Figure 2

Examples of the Chi-Square Distribution Family with Varying Degrees of Freedom

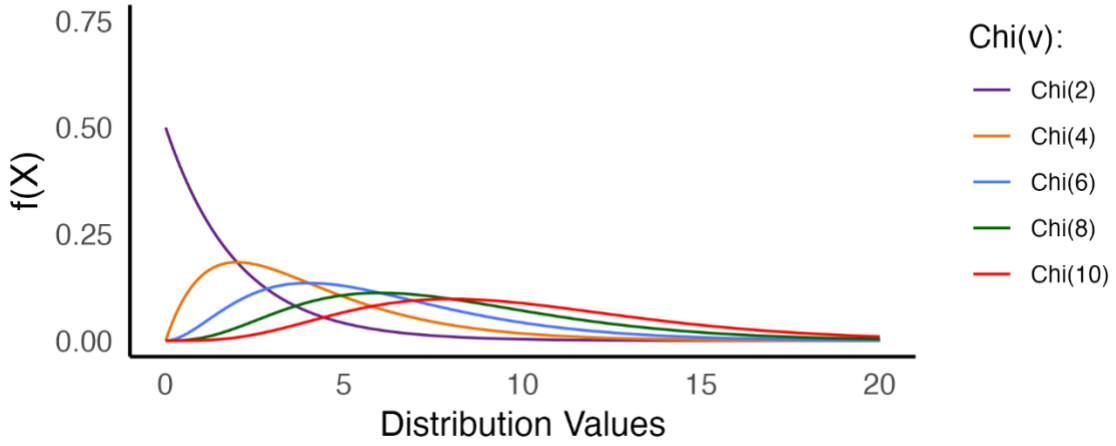


Figure 3

Examples of the Gamma Family of Distributions with Varying Hyperparameters

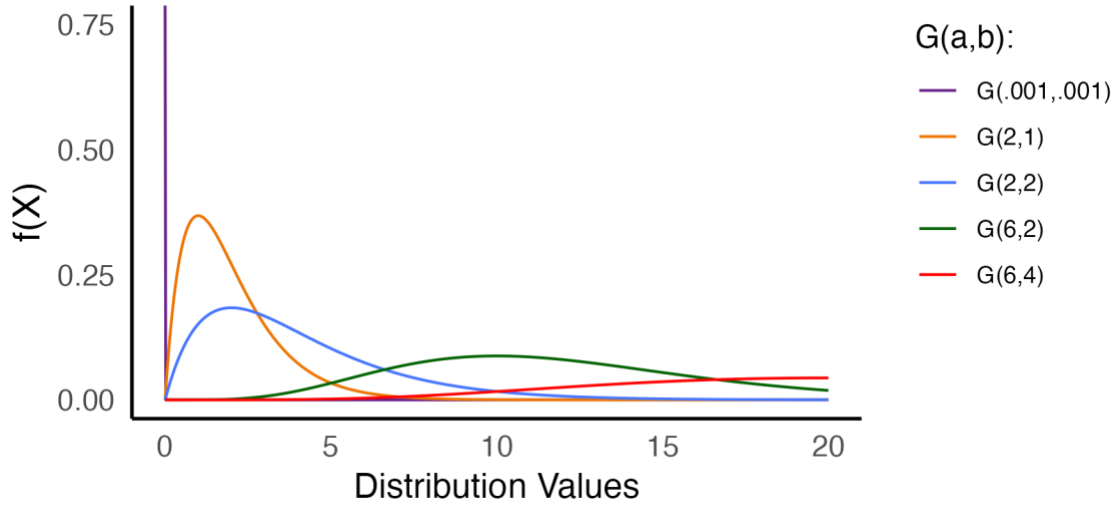


Figure 4

Examples of the Inverse Gamma Family of Distributions with Varying Hyperparameters

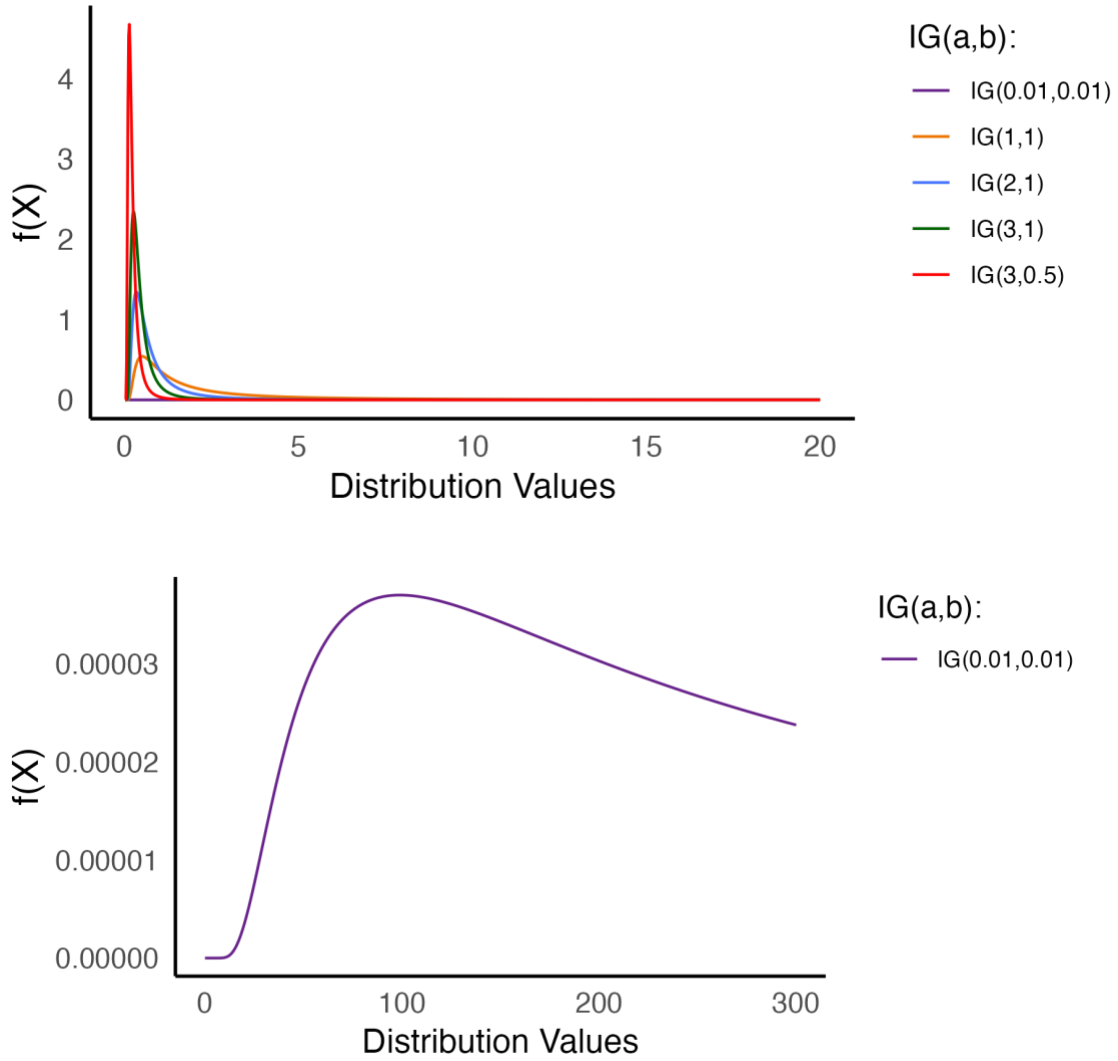


Figure 5

Examples of the Normal Family of Distributions with Varying Hyperparameters

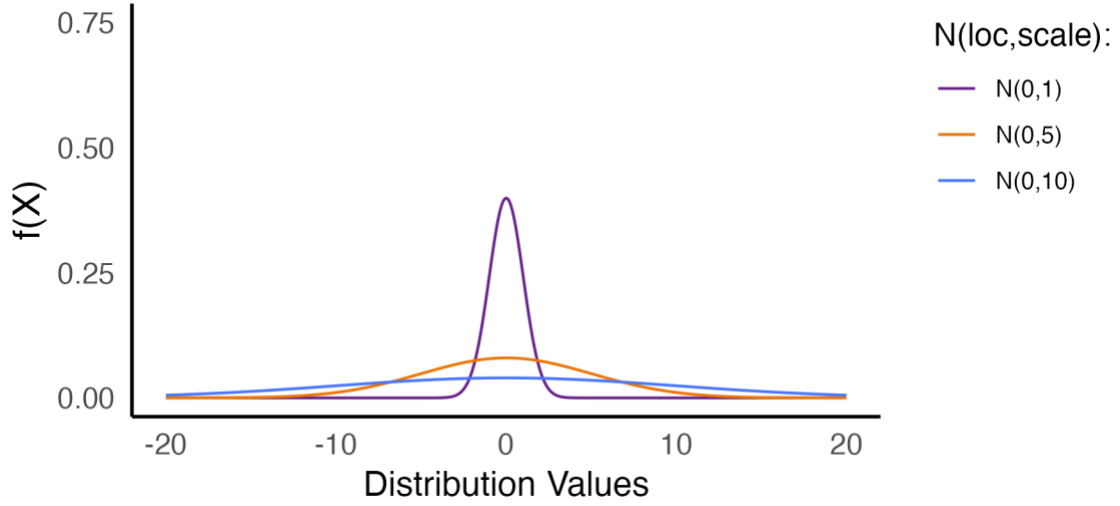


Figure 6

Examples of the Half-t Family of Distributions with Varying Hyperparameters

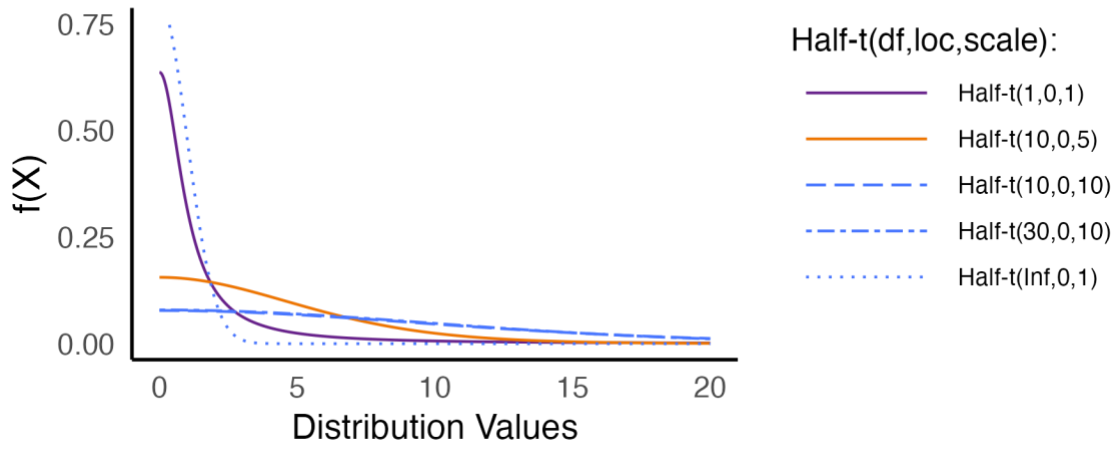


Figure 7

Examples of the Half-Cauchy Family of Distributions with Varying Hyperparameters

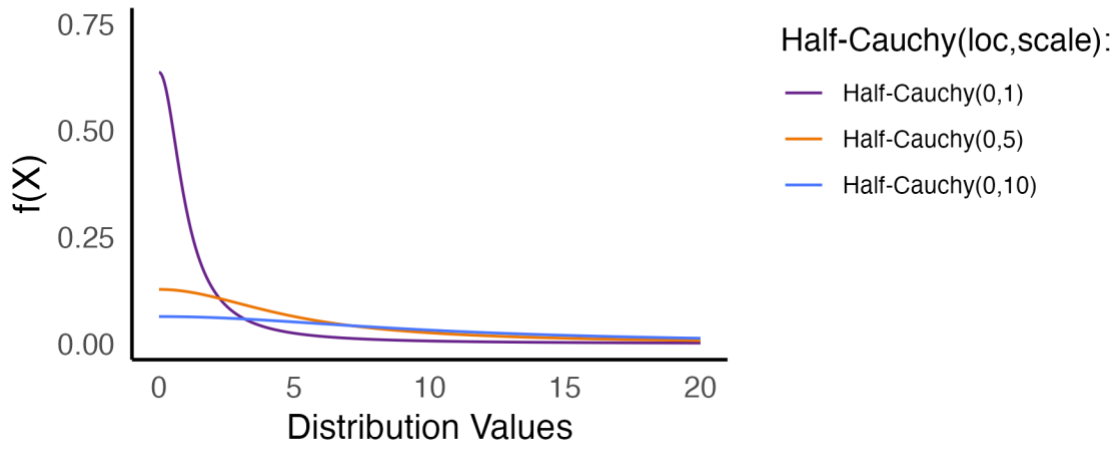


Figure 8

Examples of the Uniform Family of Distributions with Varying Hyperparameters

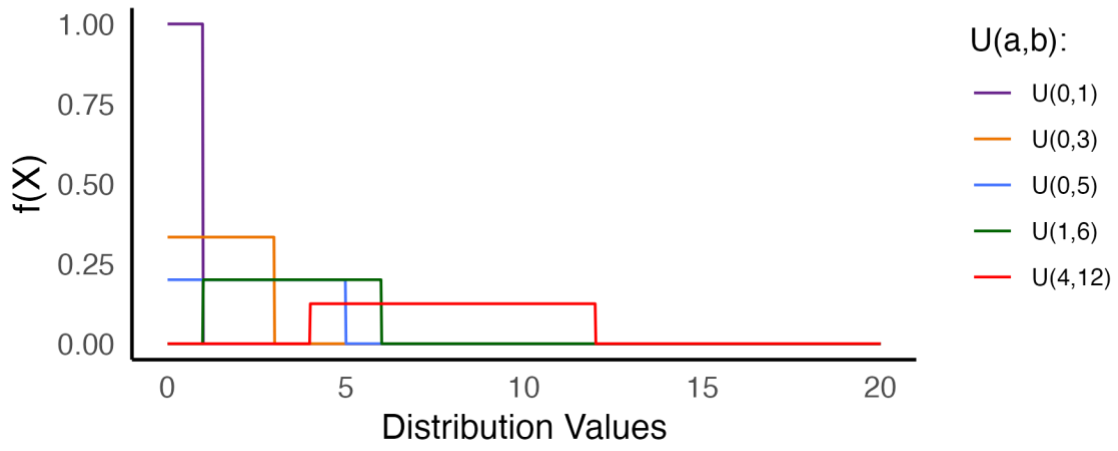
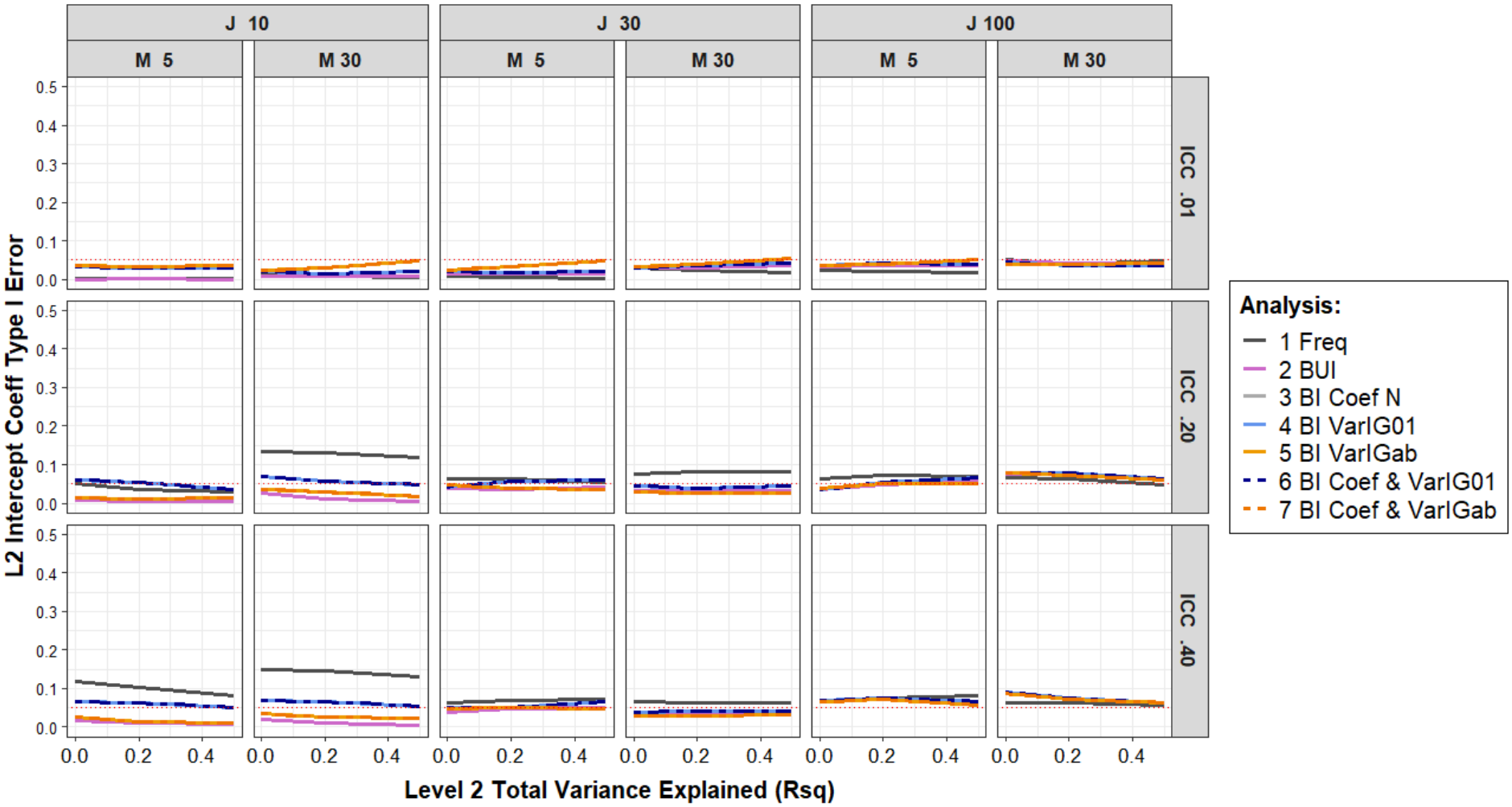


Figure 9

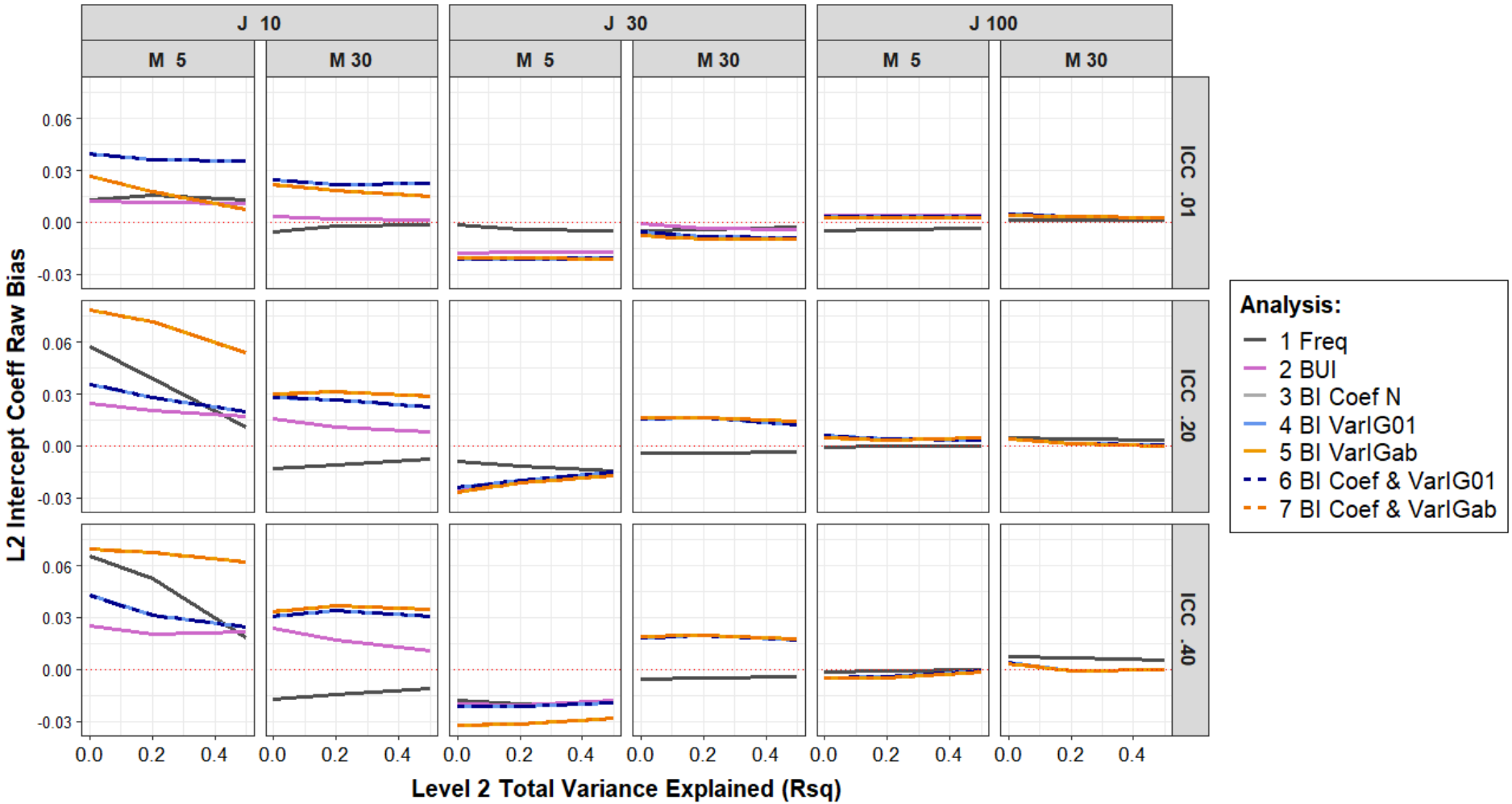
False Positive Rates for L2 Intercept Coefficient by Design Condition, Collapsed across L1 Predictor Effects



Note. False Positive rates are for all x-axis levels because intercept was set to zero (nominal false positive rate is .05). Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 10

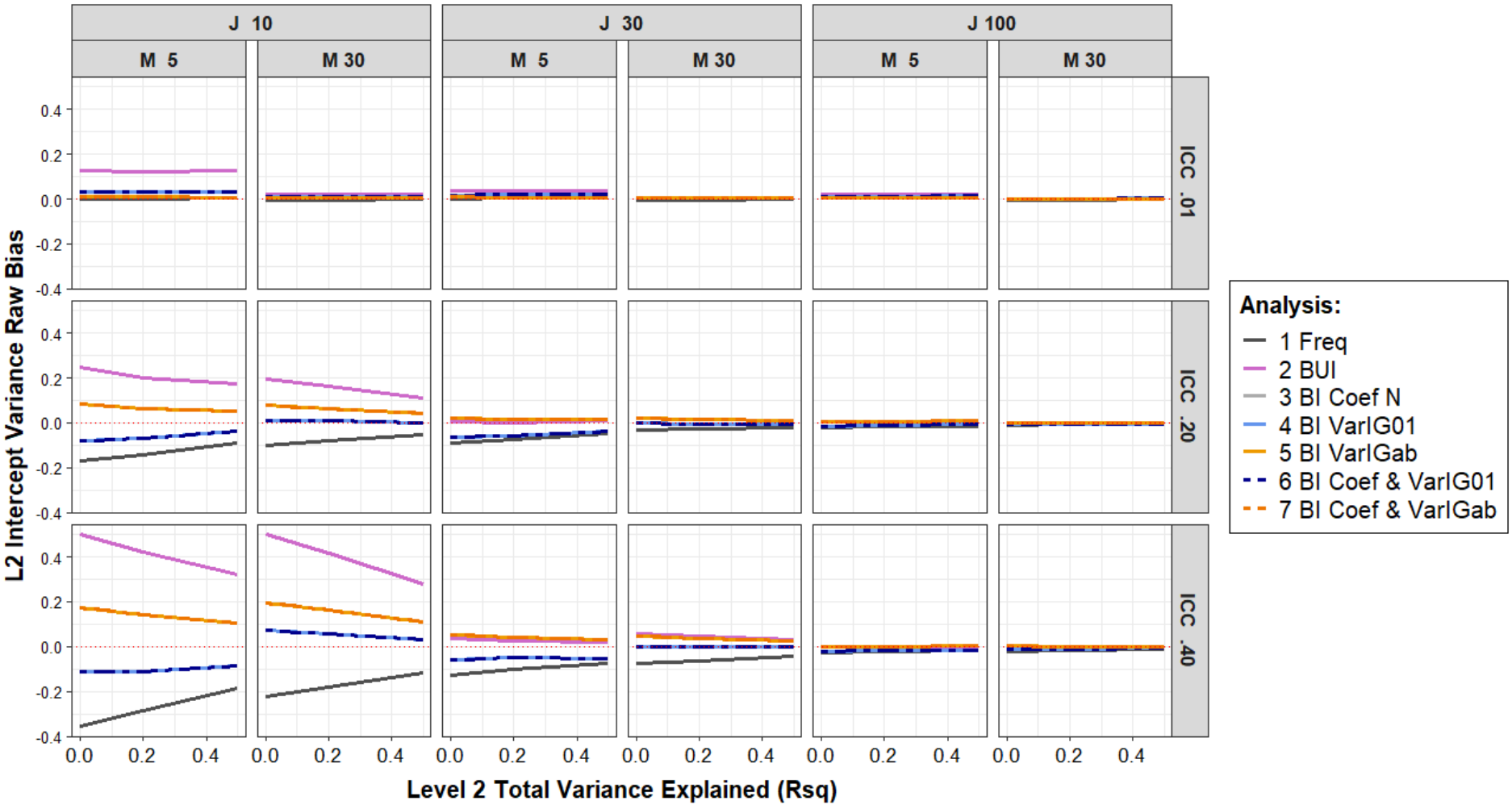
Raw Bias Results for L2 Intercept Coefficient by Design Condition, Collapsed across L1 Predictor Effects



Note. Raw bias is difference between observed and population values. Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 11

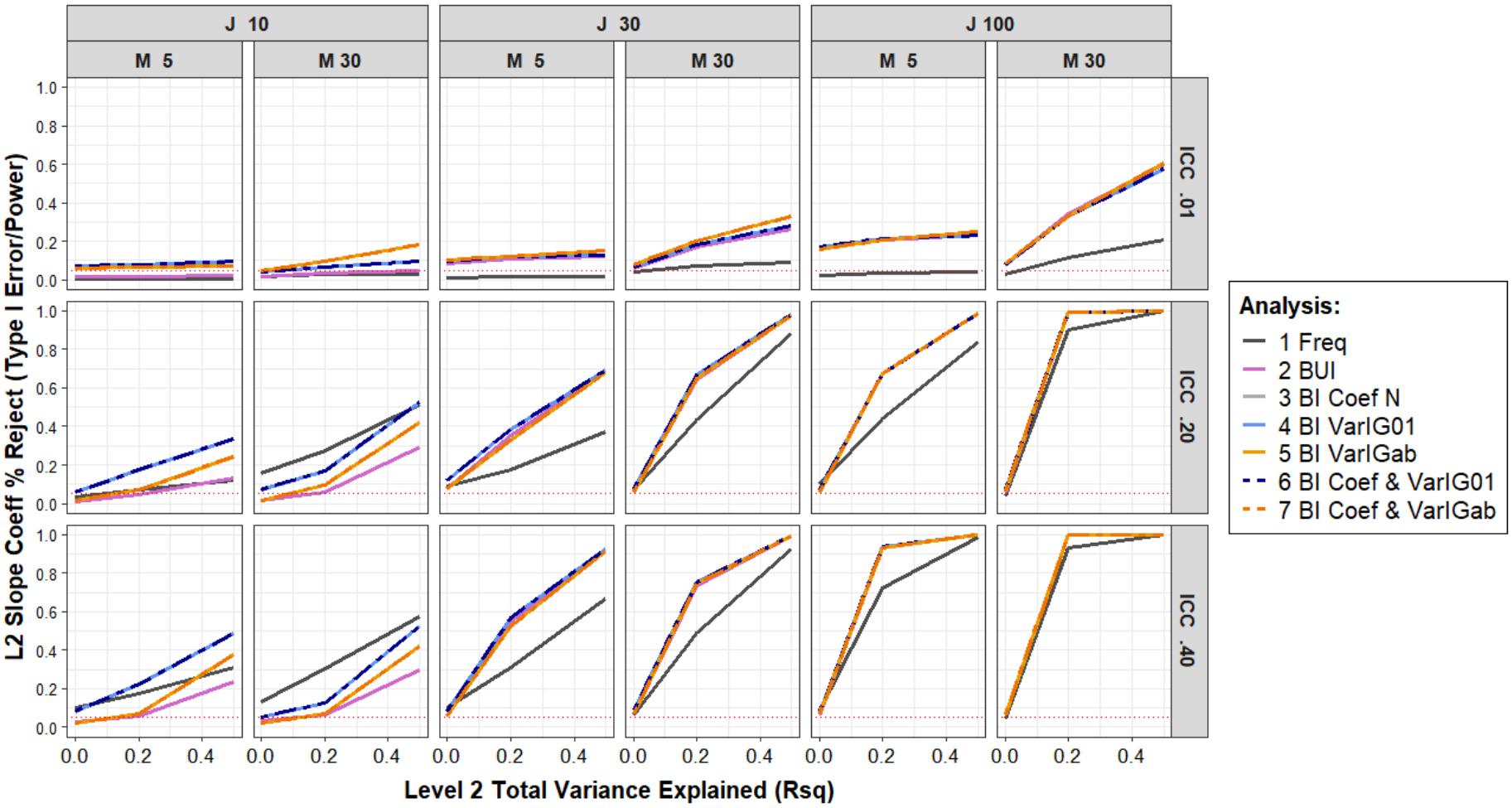
Raw Bias Results for L2 Intercept Residual Variance by Design Condition, Collapsed across L1 Predictor Effects



Note. Raw bias is difference between observed and population values. Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 12

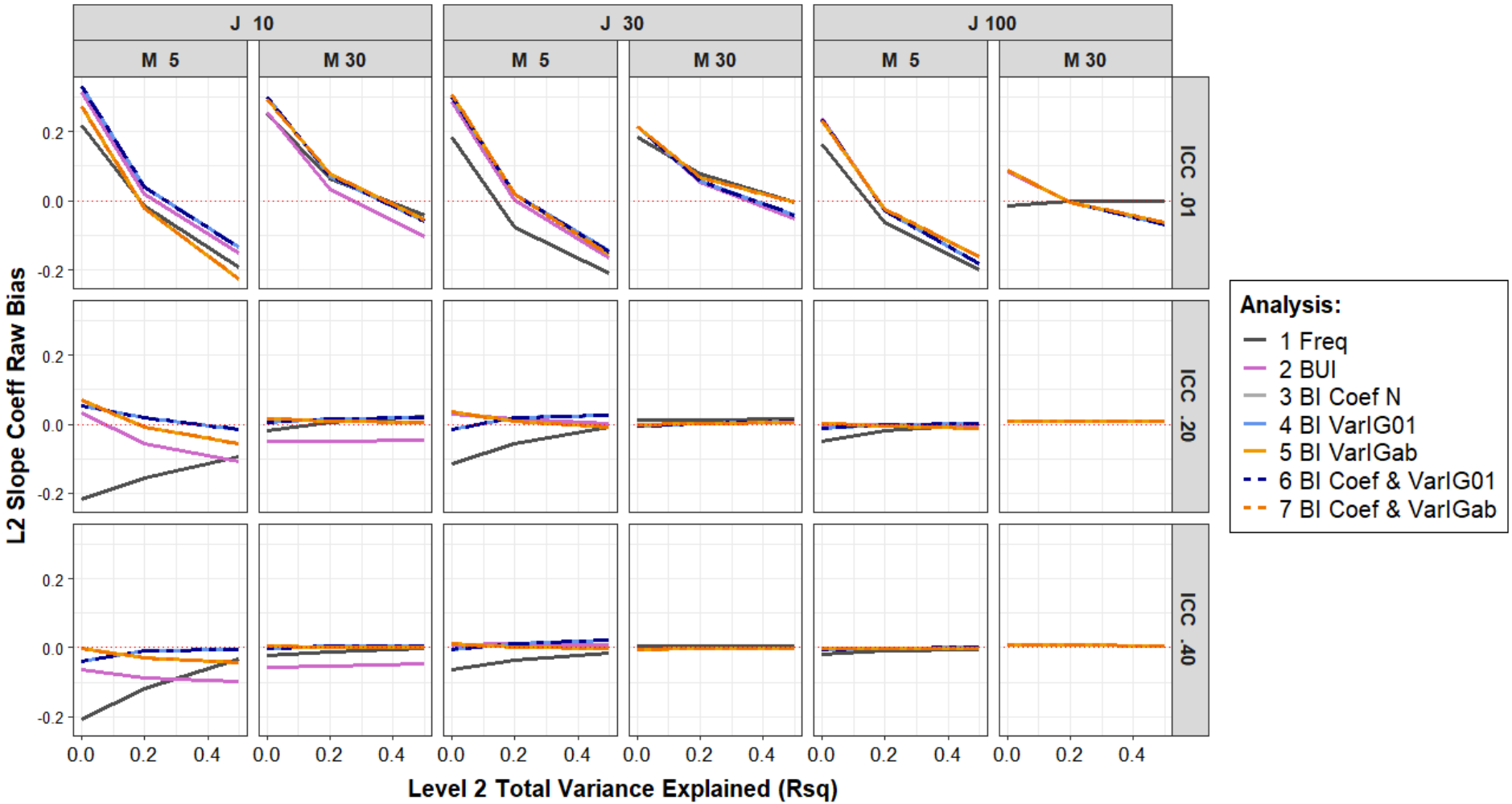
False Positive Type I Error Rates and Correct Positive Power for L2 Predictor Slope Coefficient by Design Condition, Collapsed across L1 Predictor Effects



Note. False positive shown for values of 0 on *x*-axis (nominal level .05); Correct positive shown for values > 0 on *x*-axis. Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 13

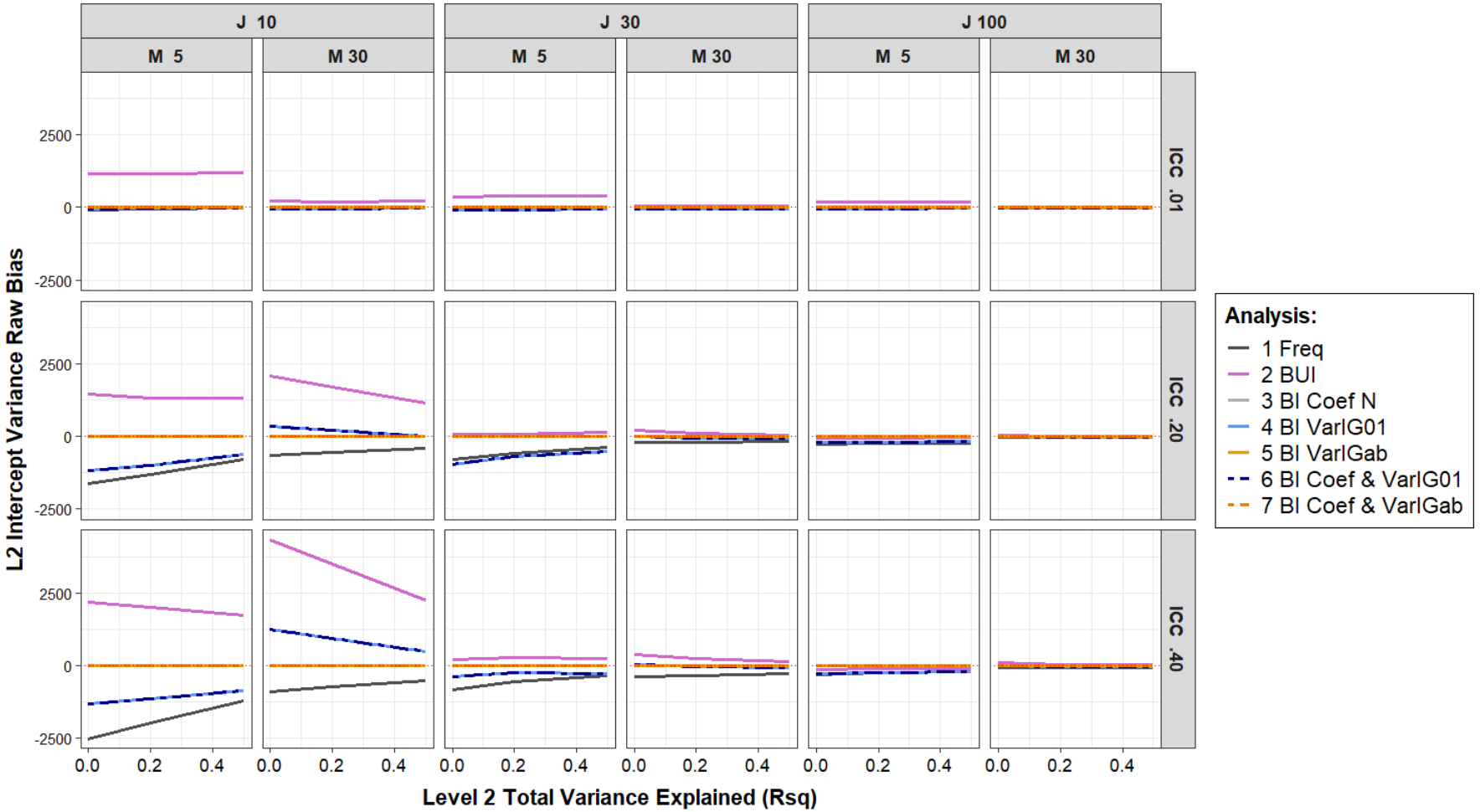
Raw Bias Results for L2 Predictor Coefficient by Design Condition, Collapsed across L1 Predictor Effects



Note. Raw bias is difference between observed and population values. Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 14

L2 Variance Estimate Bias for N = 20 replications for Scaled Data with True SD = 100



Note. Results for *X* and *Y* variables scaled to have total variance = 10,000. Raw bias is difference between observed and population values. Freq = Frequentist, BUI = Bayesian uninformative (flat) priors, BI = Bayesian with informative priors; N = Normal Distribution used for predictor coefficient priors, VarIG01 = *IG*(.01,.01) used for variance component priors, VARIGab = *IG*(a,b) used for variance component priors, with a and b set to have a mode equal to the true variance component value.

Figure 15

Chi-square, Gamma and Inverse Gamma Distributions with Full Sample

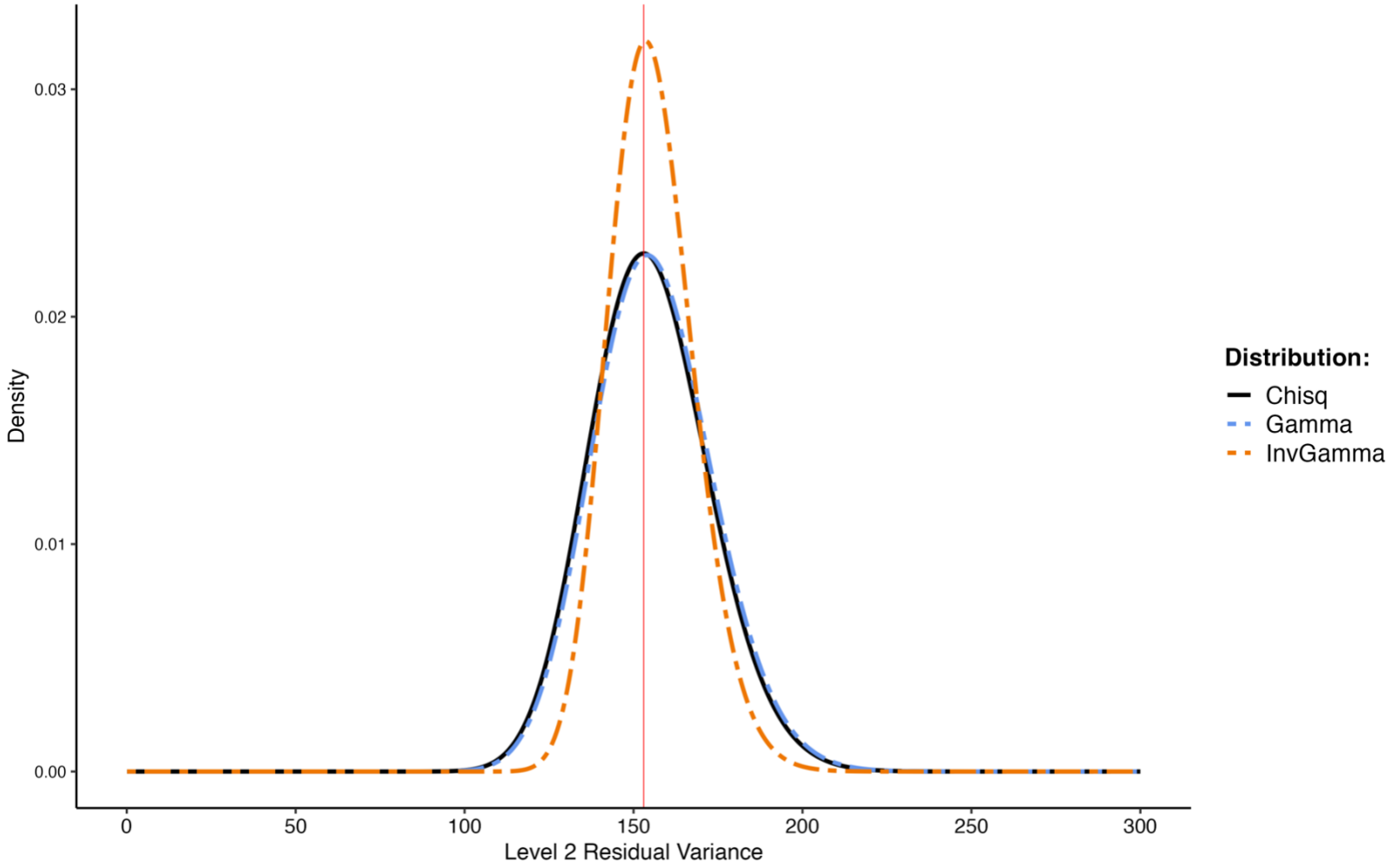


Figure 16

Chi-square, Gamma and Inverse Gamma Distributions with Subsample

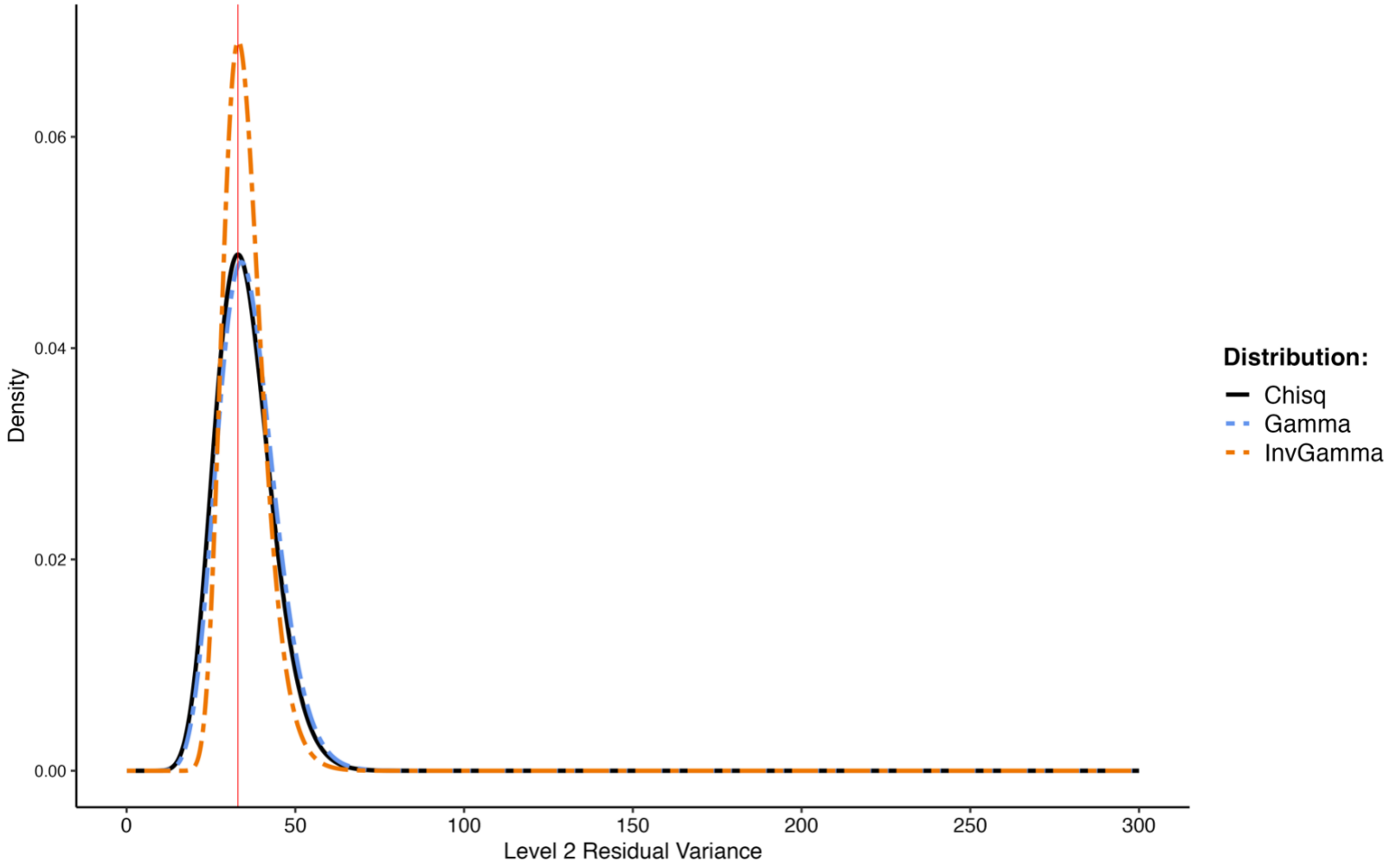
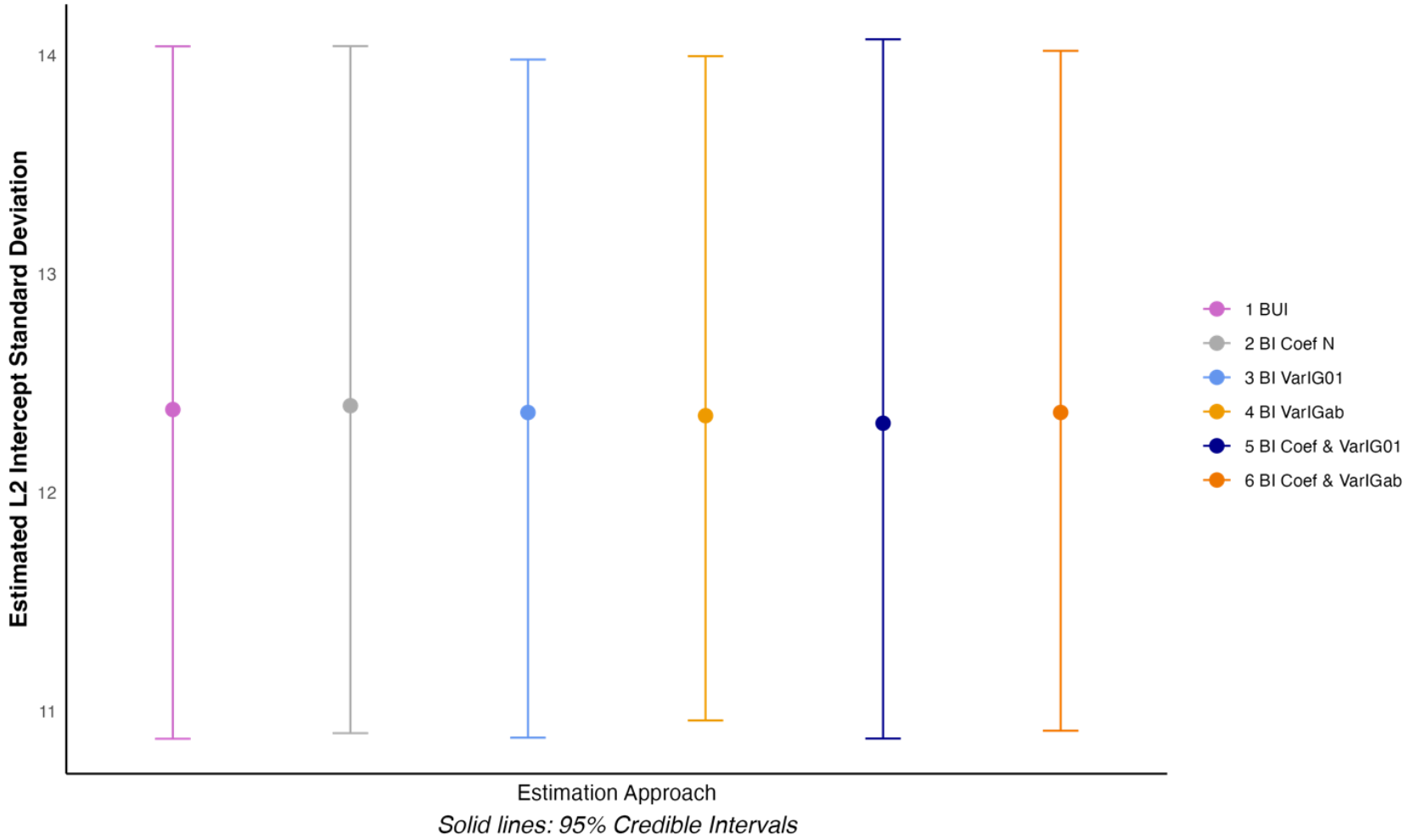


Figure 17

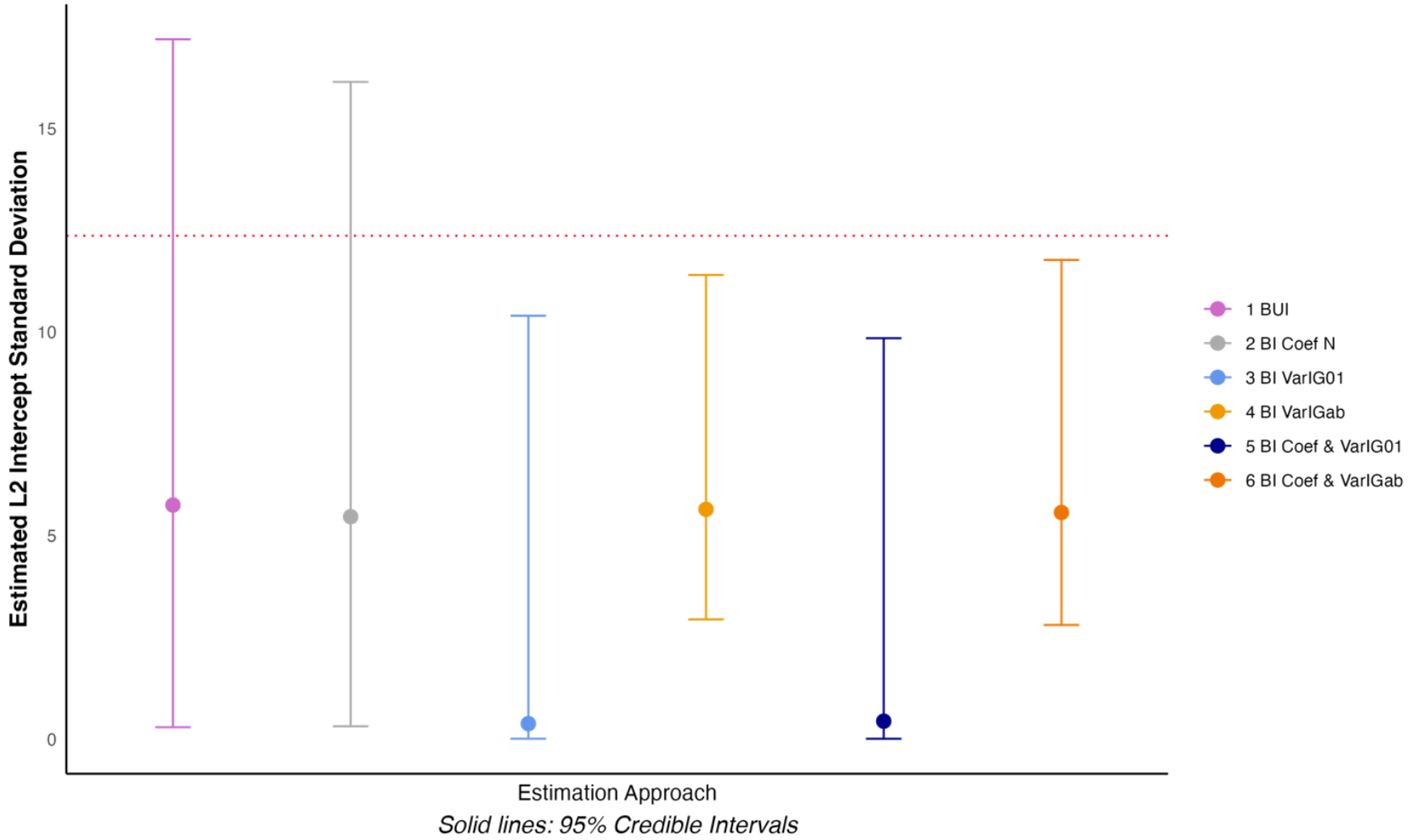
Estimated L2 Intercept Standard Deviation Using Inverse Gamma Prior Informativeness (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 18

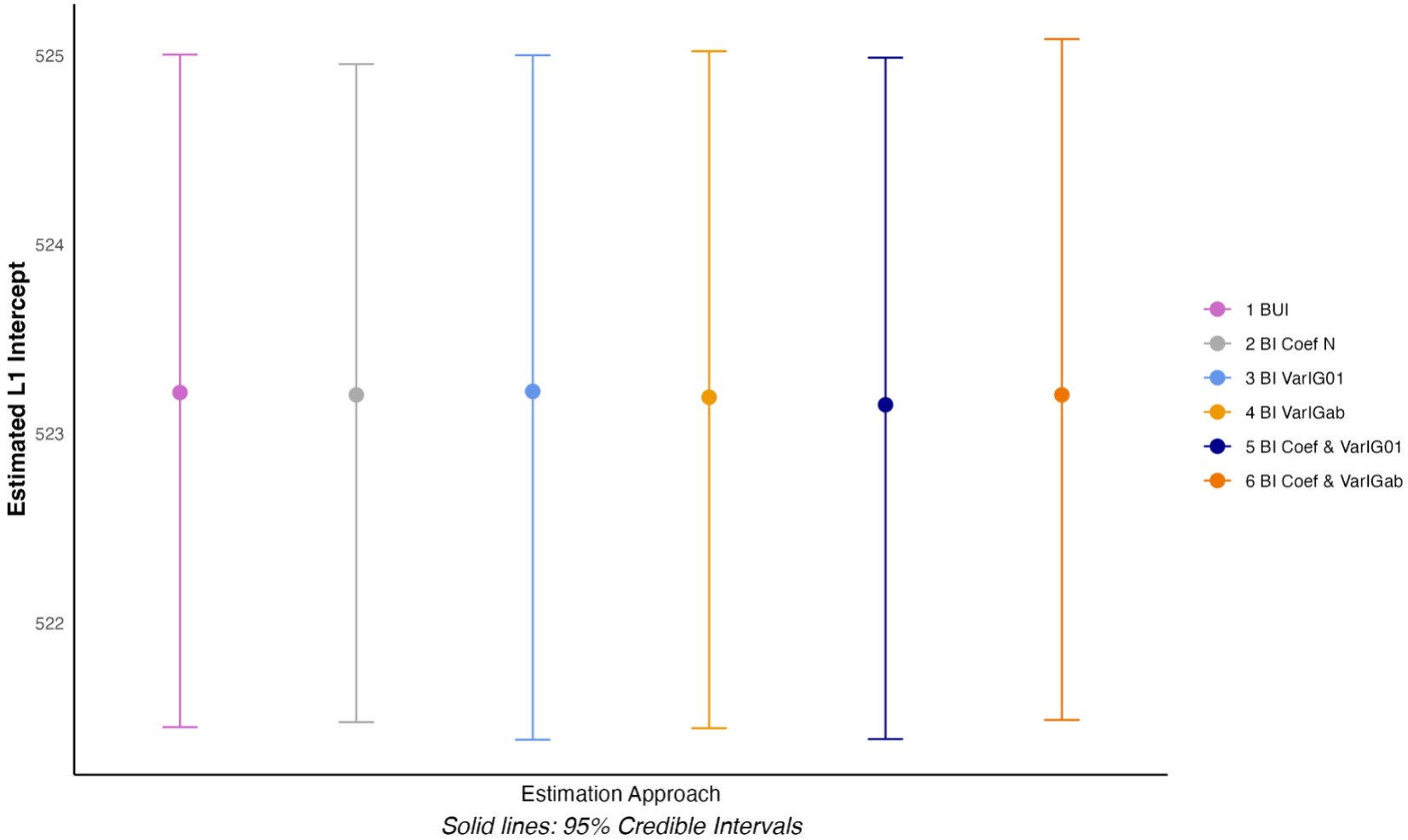
Estimated L2 Intercept Standard Deviation Using Inverse Gamma Prior Informativeness (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 19

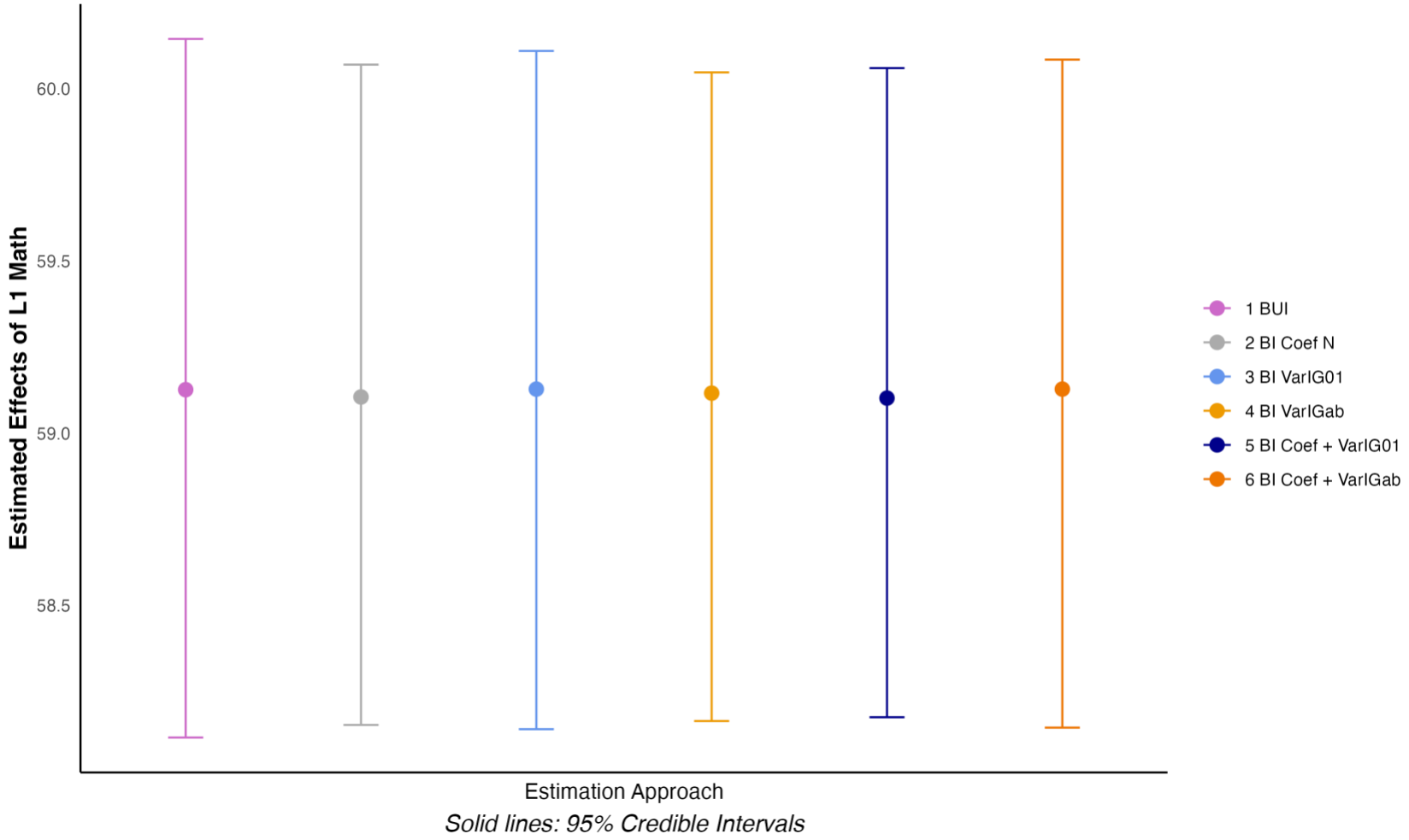
Estimating L1 Intercept Using Inverse Gamma Prior Informativeness (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 20

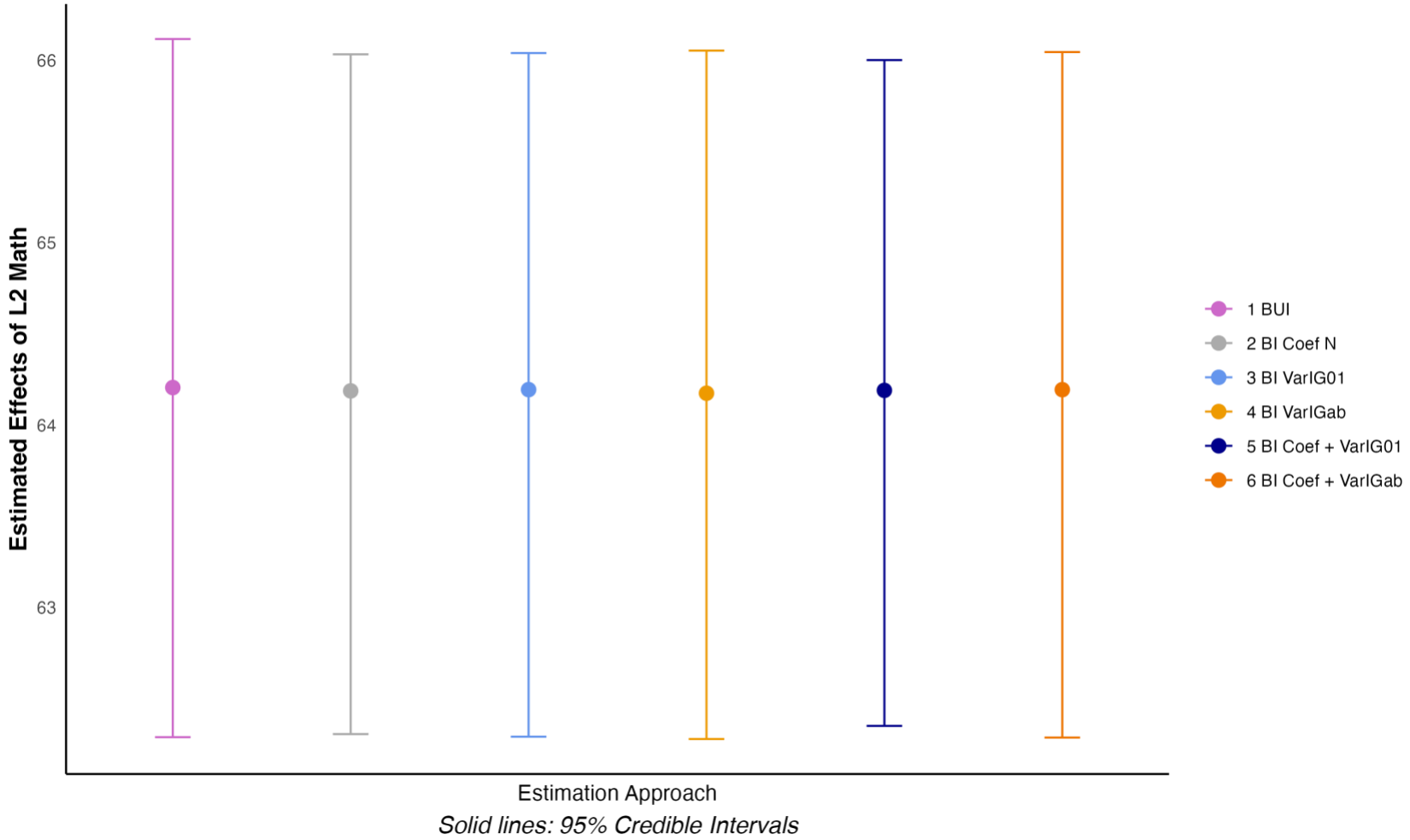
Estimating L1 Math (Centered, Z) Effects Using Inverse Gamma Prior Informativeness (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 21

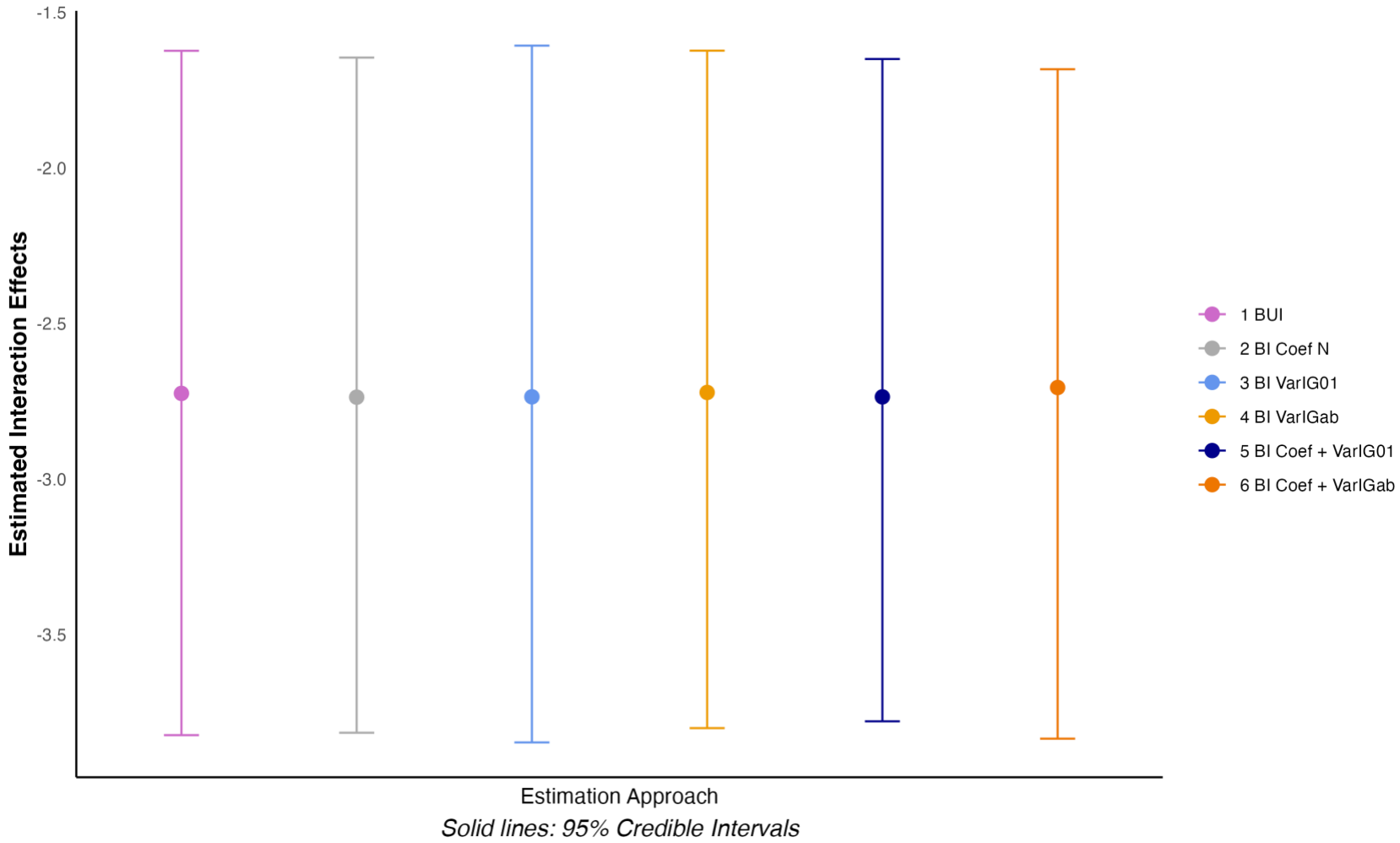
Estimating L2 Math (Aggregate, Z) Effects Using Inverse Gamma Prior Informativeness (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 22

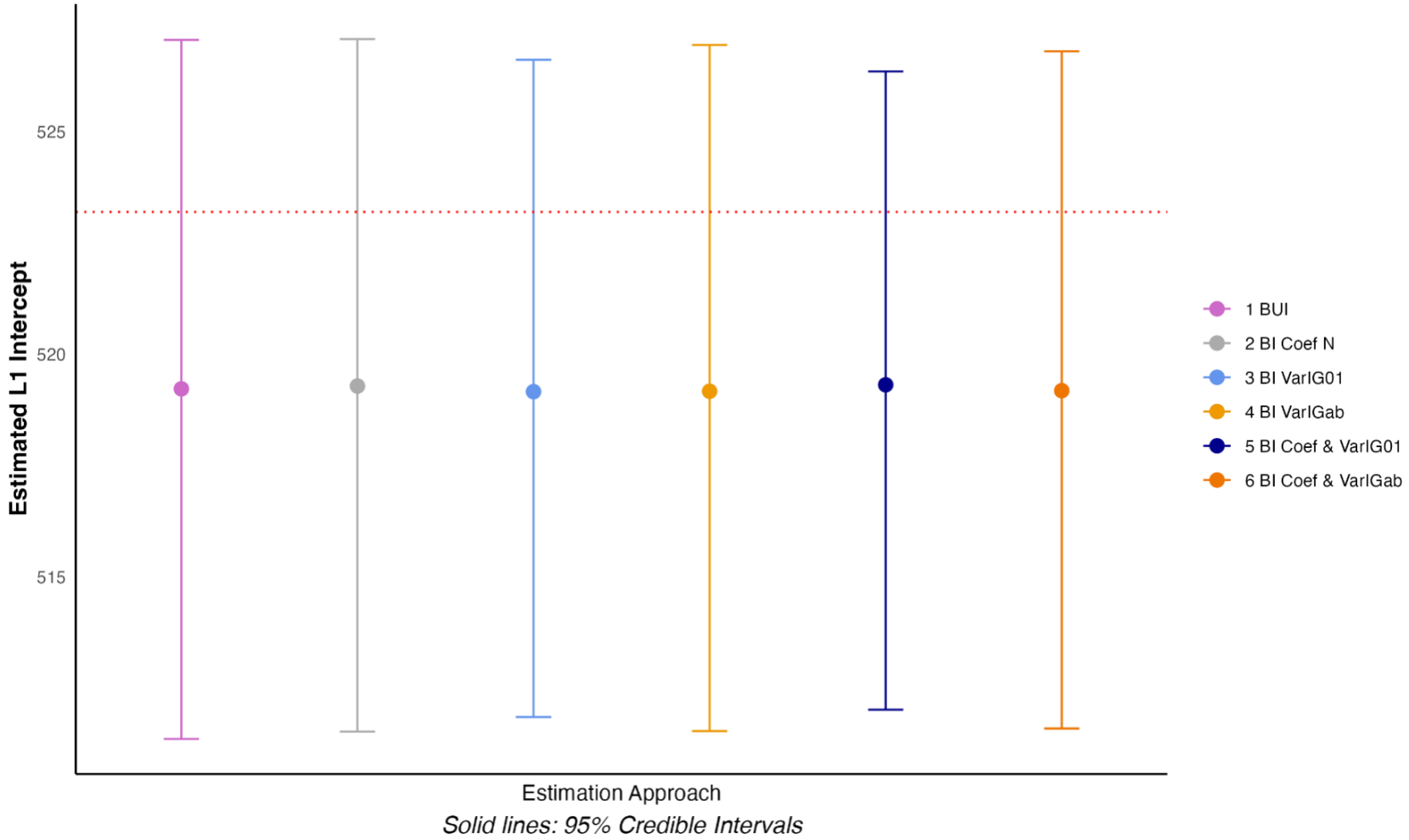
Estimating Interaction Effects of L1 and L2 Math Using Inverse Gamma Prior Informativeness (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 23

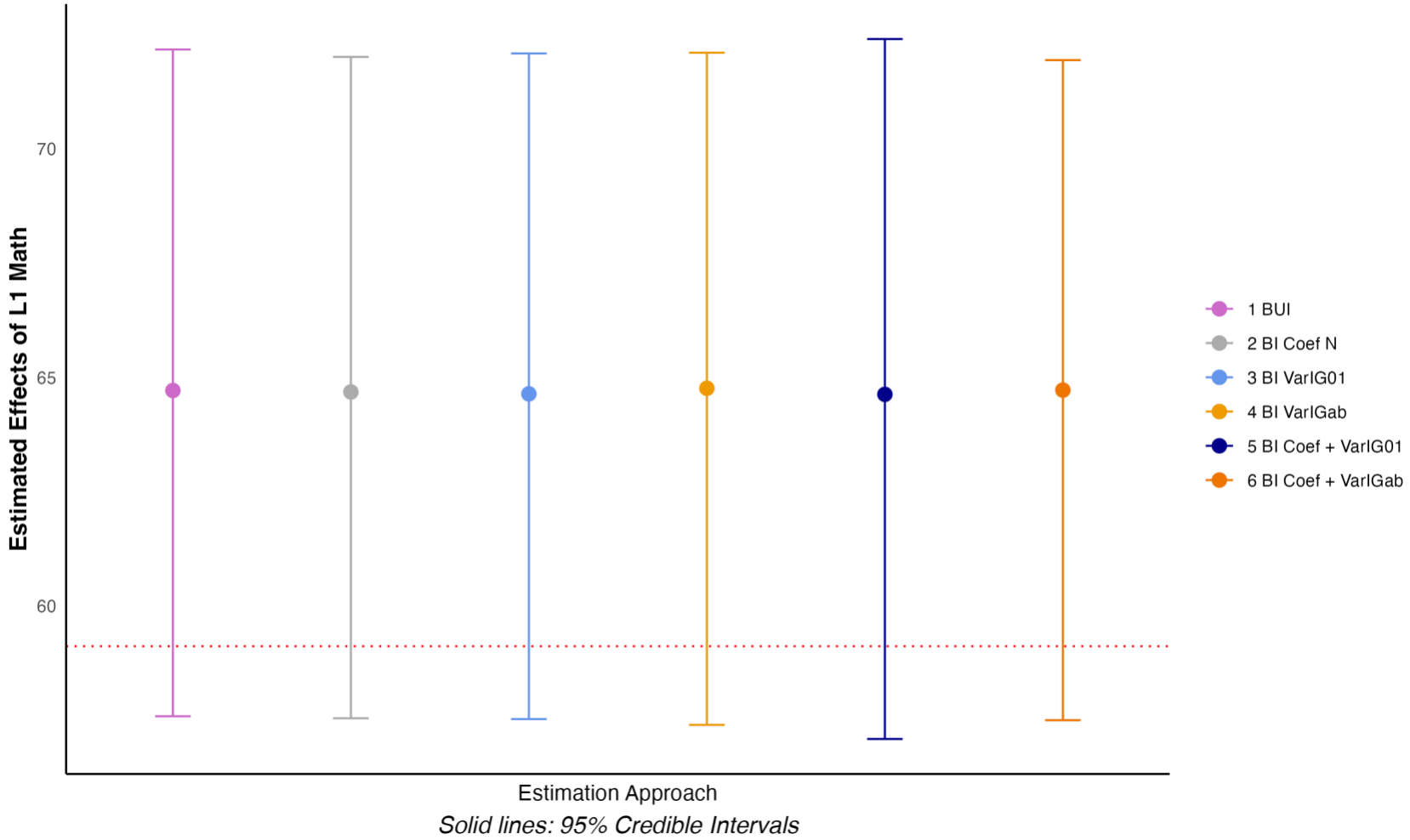
Estimating L1 Intercept Using Inverse Gamma Prior Informativeness (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 24

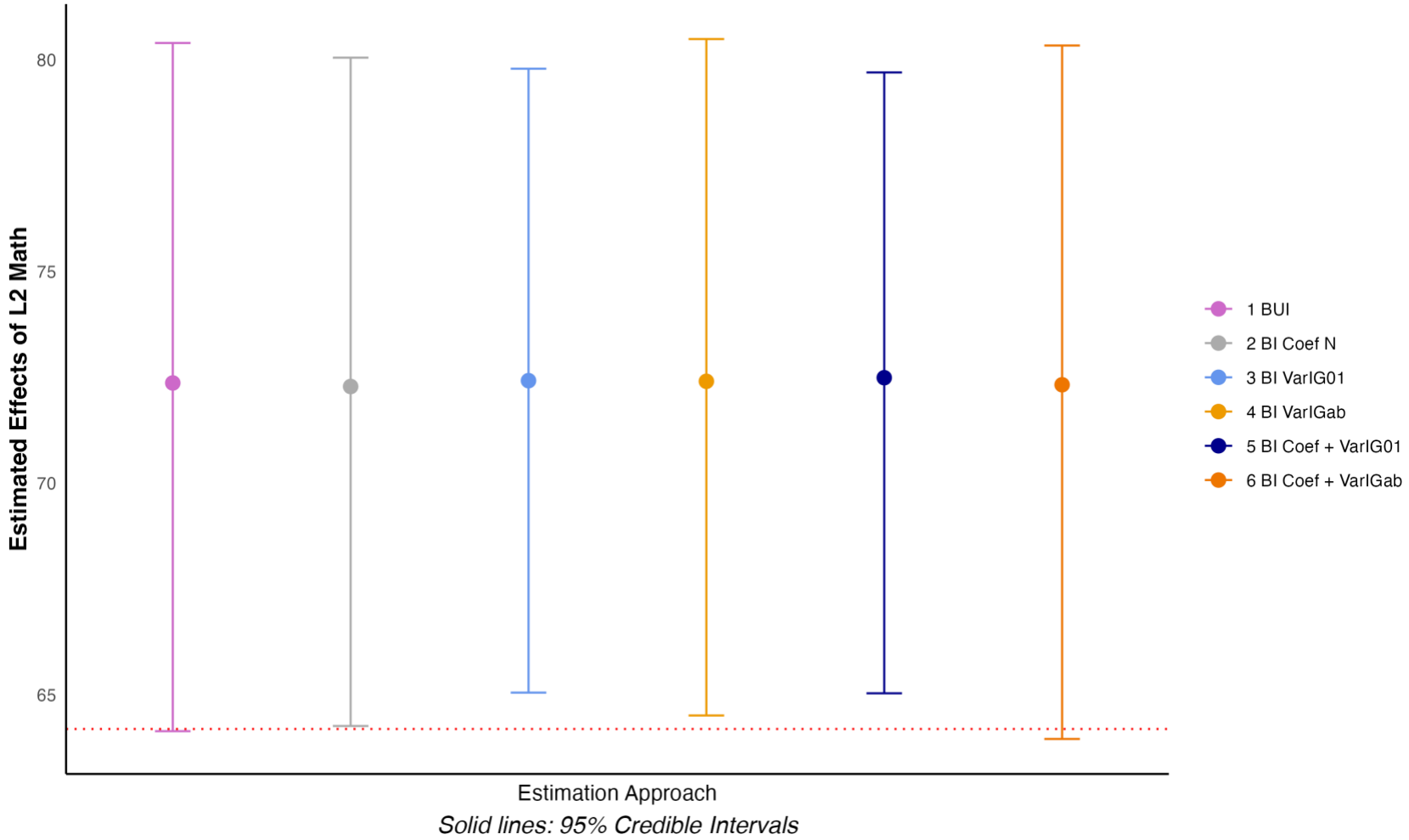
Estimating L1 Math (Centered, Z) Effects Using Inverse Gamma Prior Informativeness (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 25

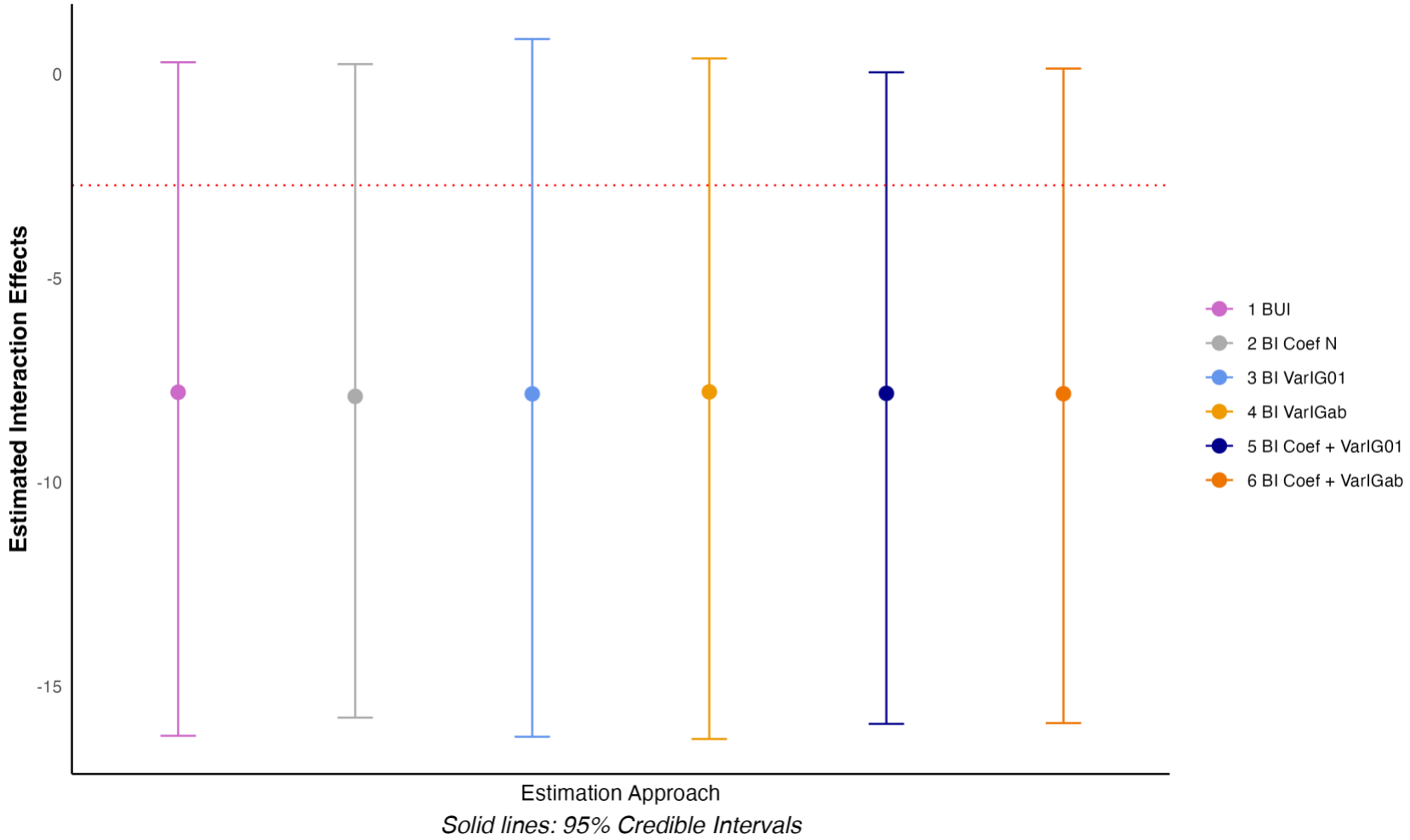
Estimating L2 Math (Aggregate, Z) Effects Using Inverse Gamma Prior Informativeness (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 26

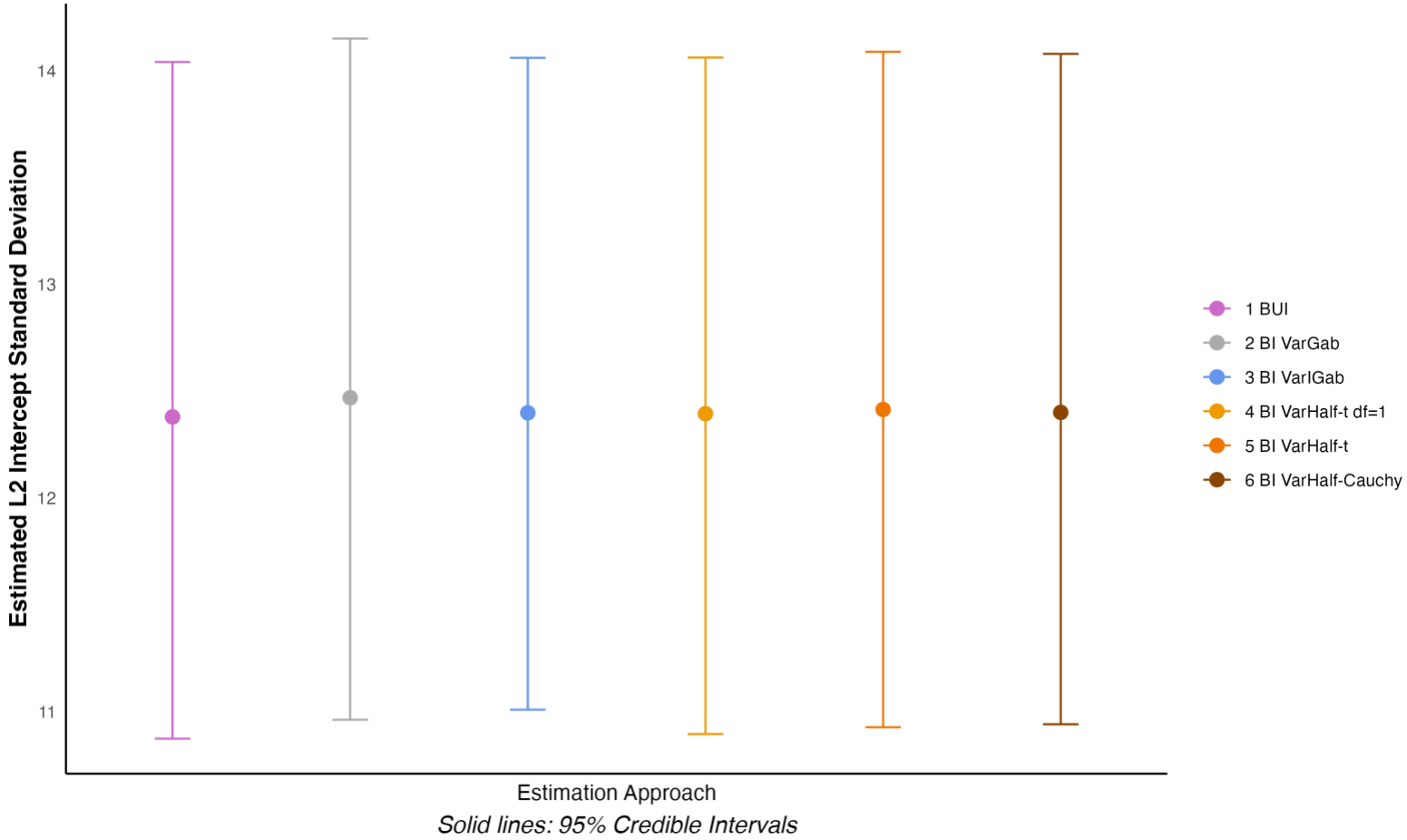
Estimating Interaction Effects of L1 and L2 Math Using Inverse Gamma Prior Informativeness (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. N = Normal distribution used for predictor coefficient priors, VarIG01 = Inverse gamma *IG*(.01,.01) used for variance component priors, VARIGab = Inverse gamma *IG*(a,b) priors used for variance components; a and b are initially calculated from chi-square distributions, transformed to gamma, and then to inverse gamma.

Figure 27

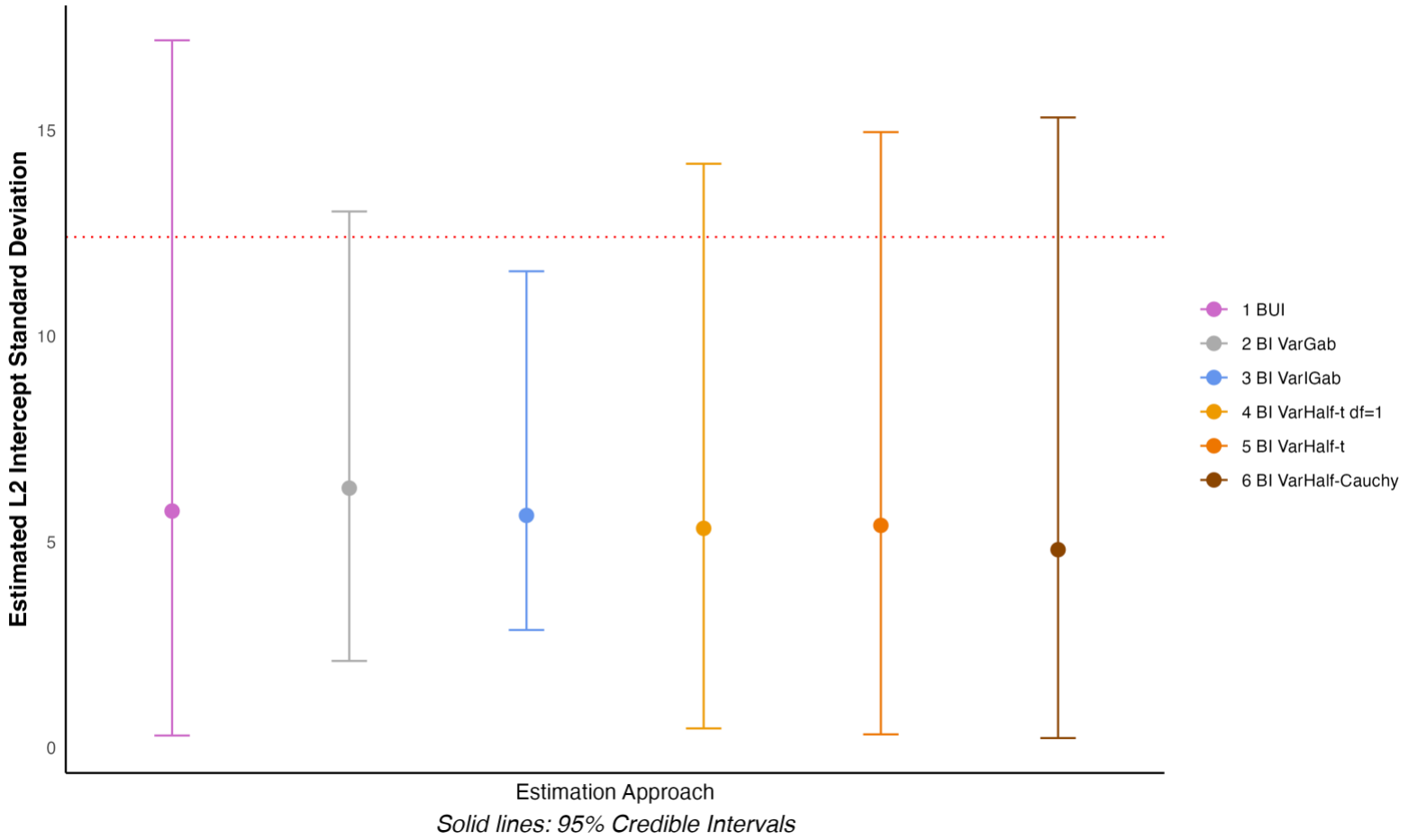
Estimated L2 Intercept Standard Deviation Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 28

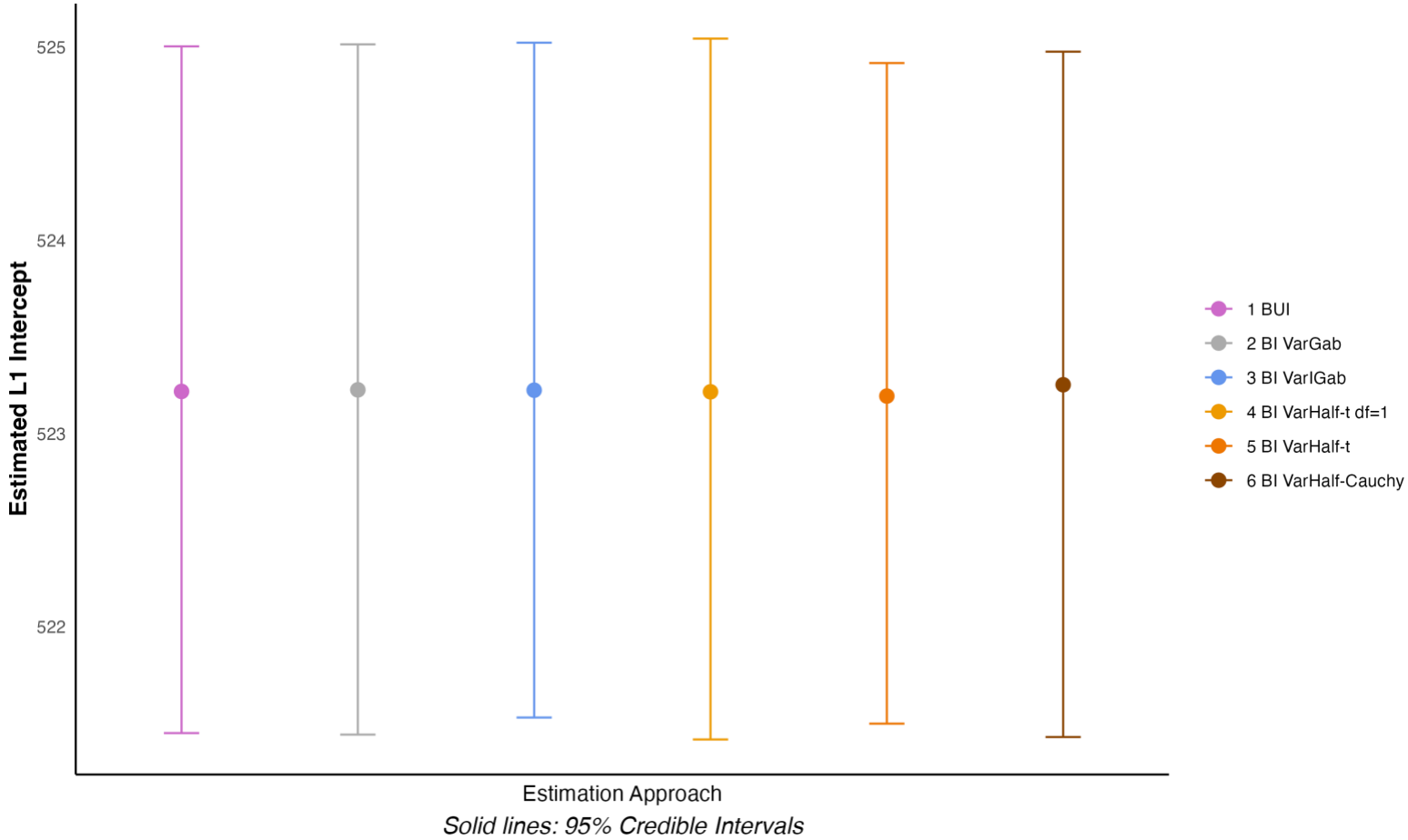
Estimated L2 Intercept Standard Deviation Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution ($df = 1$ or $df = k$). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 29

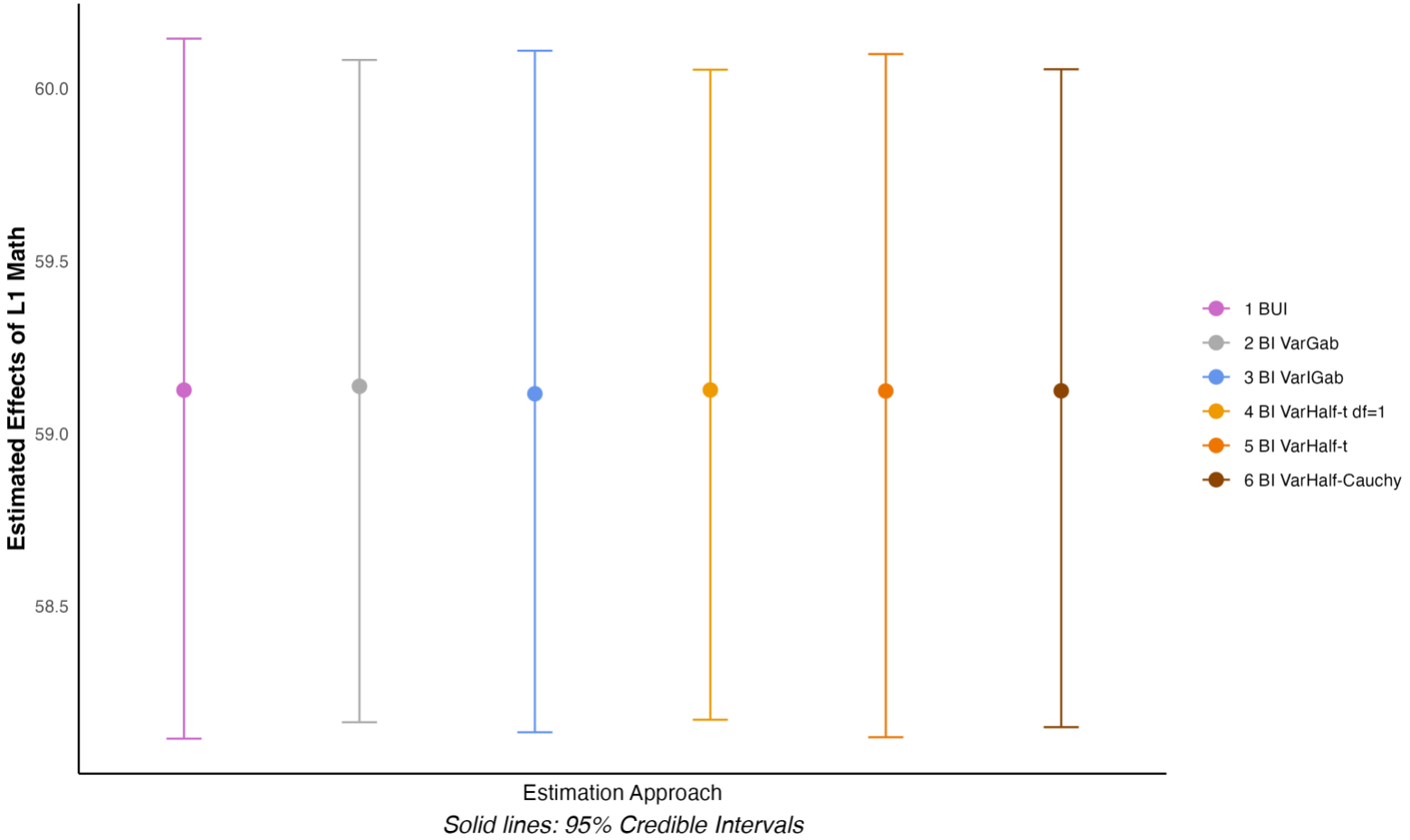
Estimating L1 Intercept Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 30

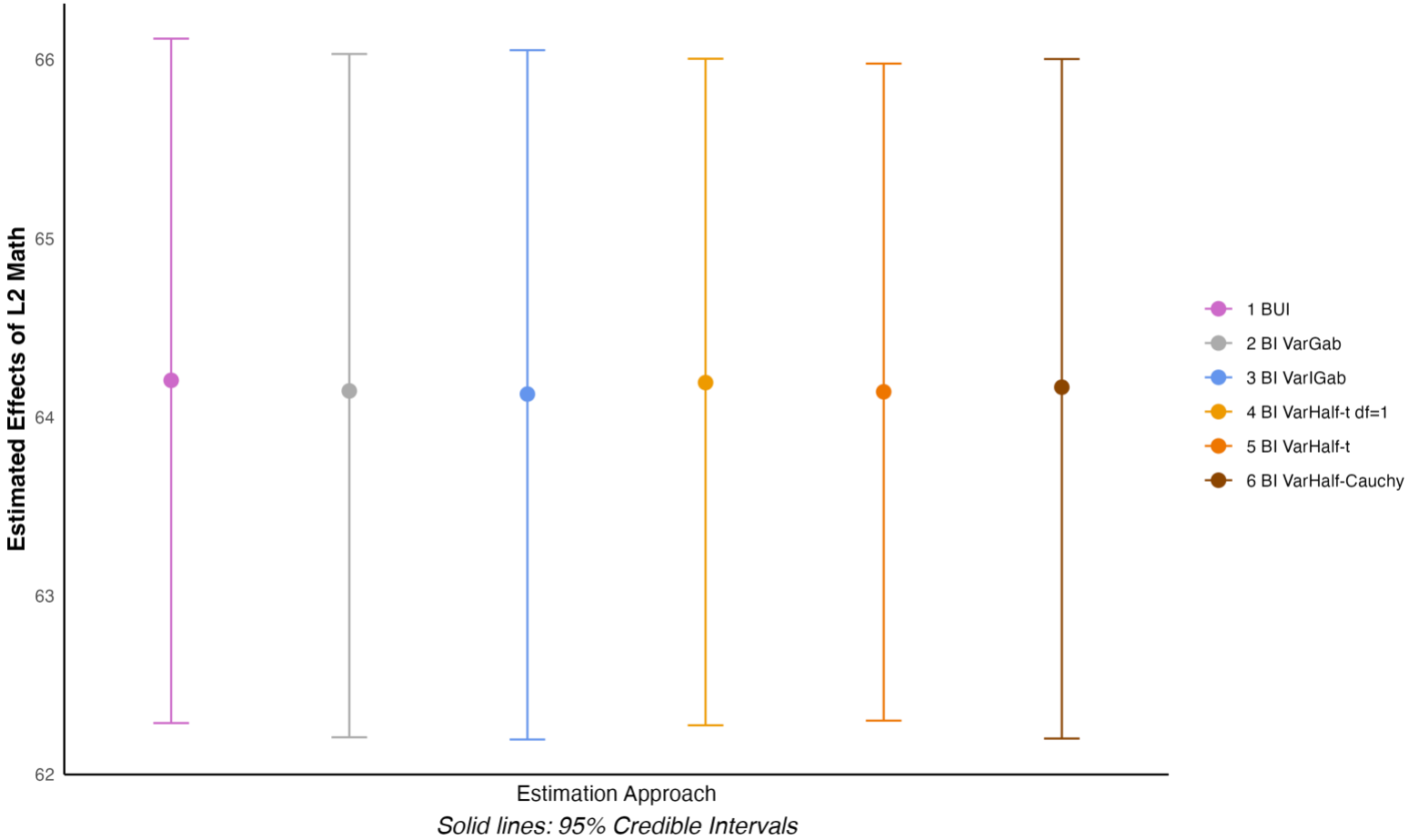
Estimating L1 Math (Centered, Z) Effects Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution ($df = 1$ or $df = k$). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 31

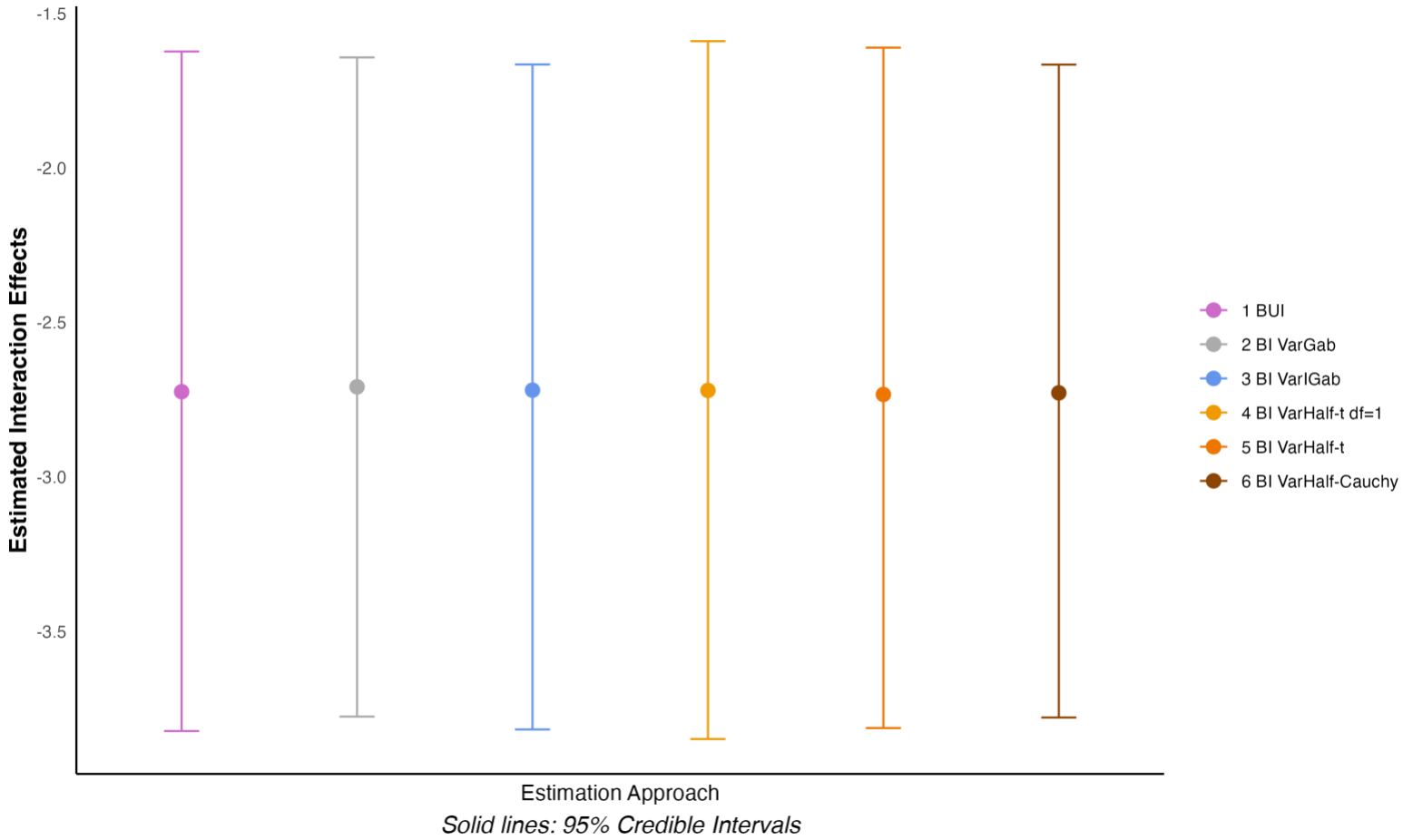
Estimating L2 Math (Aggregate, Z) Effects Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 32

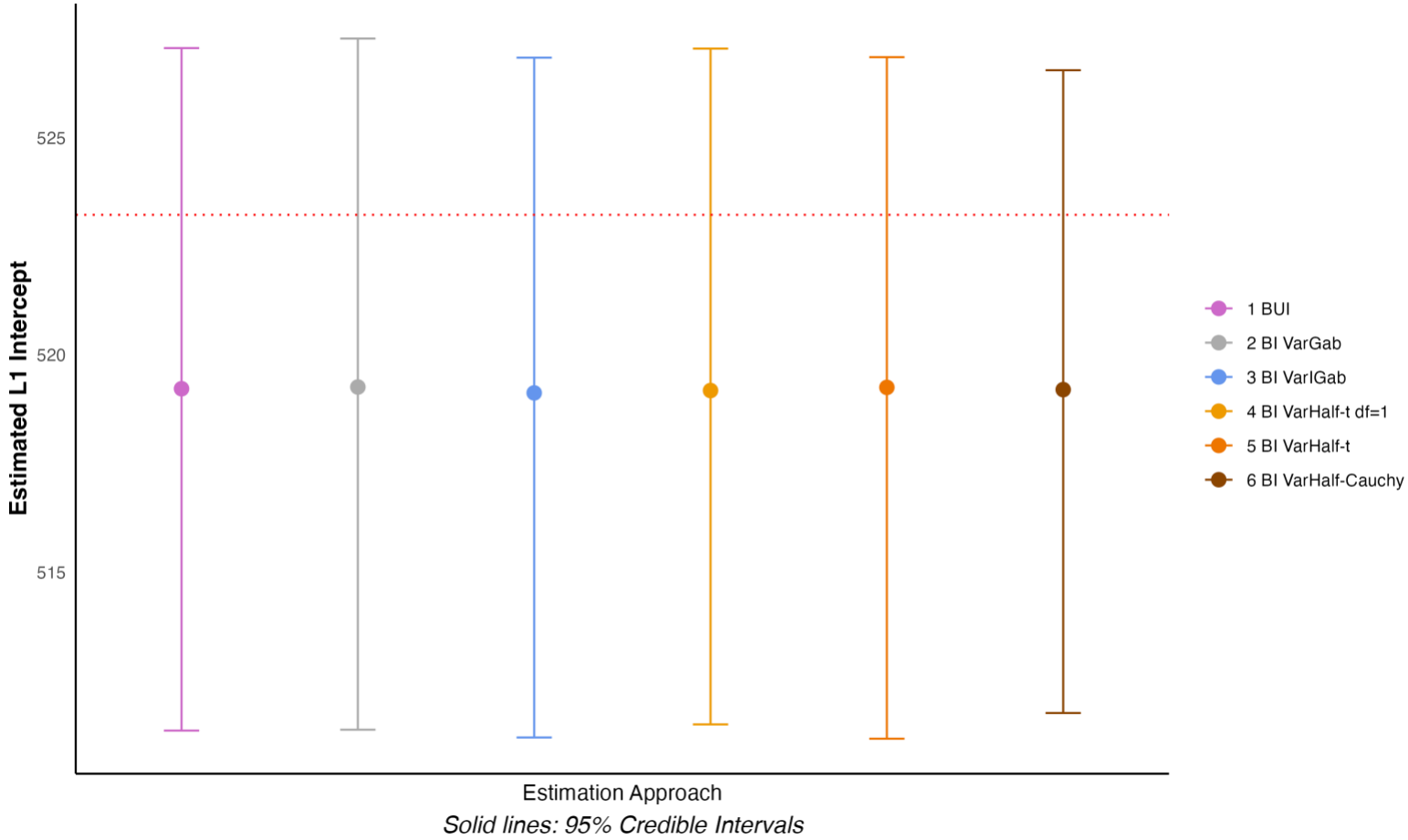
Estimating Interaction Effects of L1 and L2 Math Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Full Sample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution ($df = 1$ or $df = k$). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 33

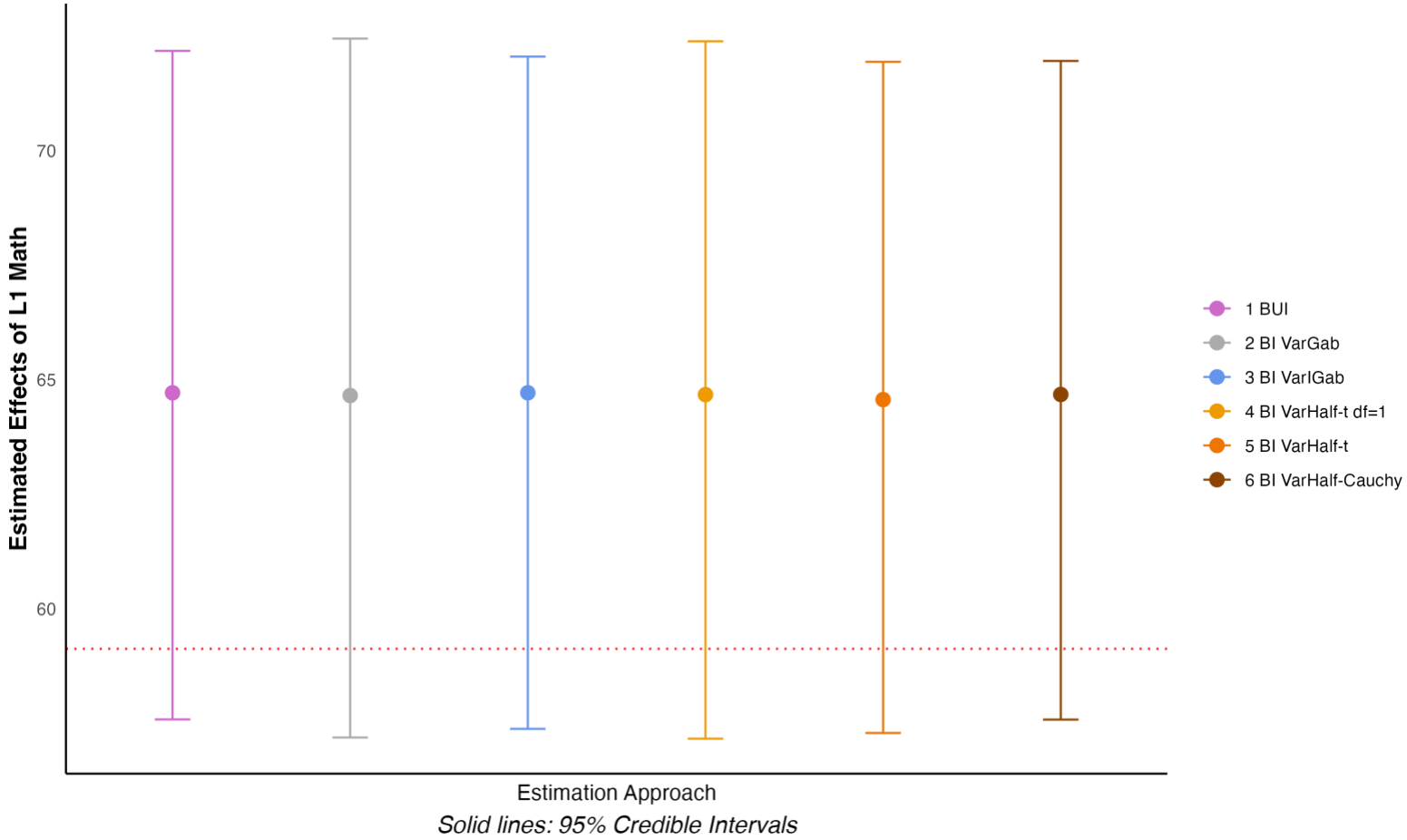
Estimating L1 Intercept Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution ($df = 1$ or $df = k$). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 34

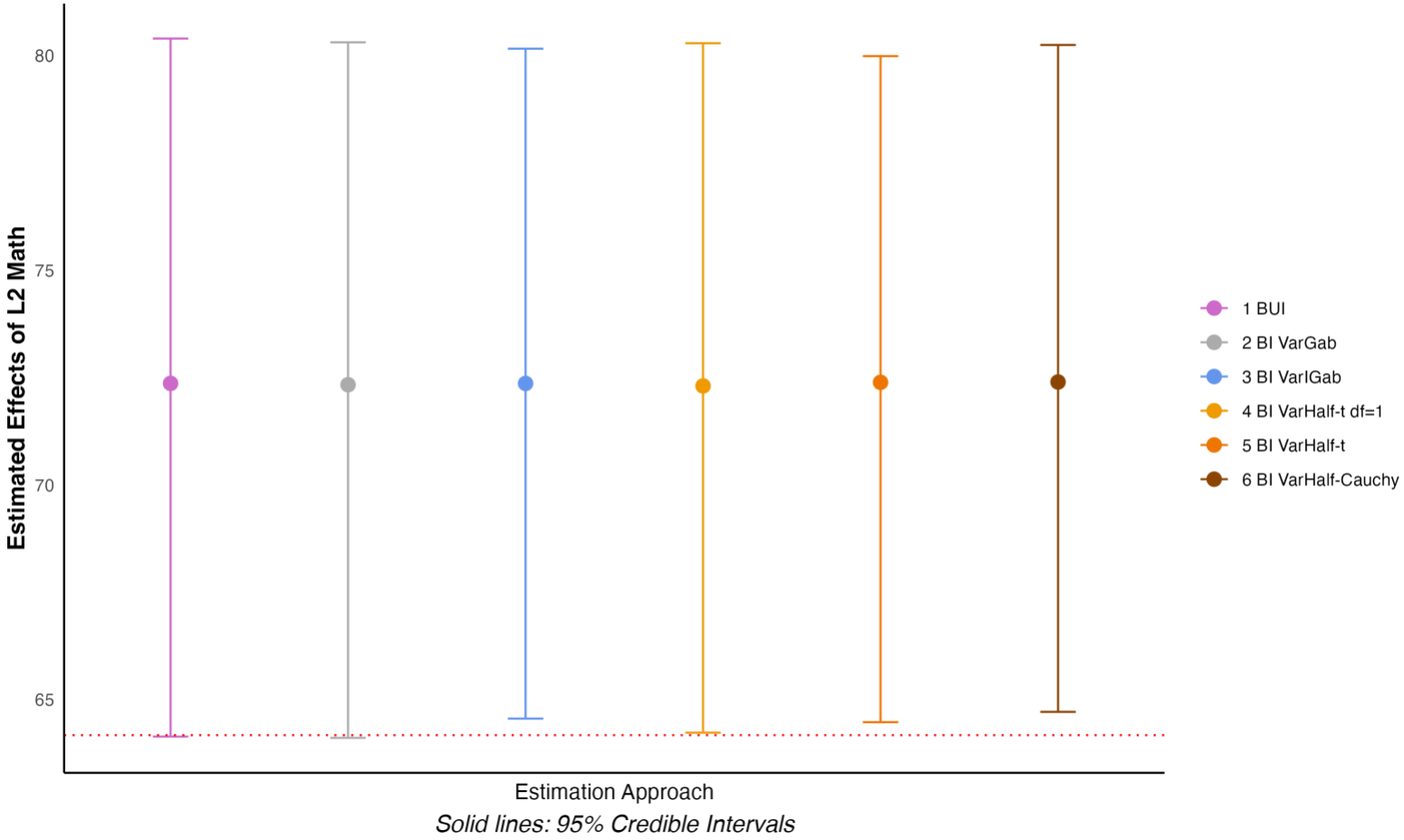
Estimating L1 Math (Centered, Z) Effects Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 35

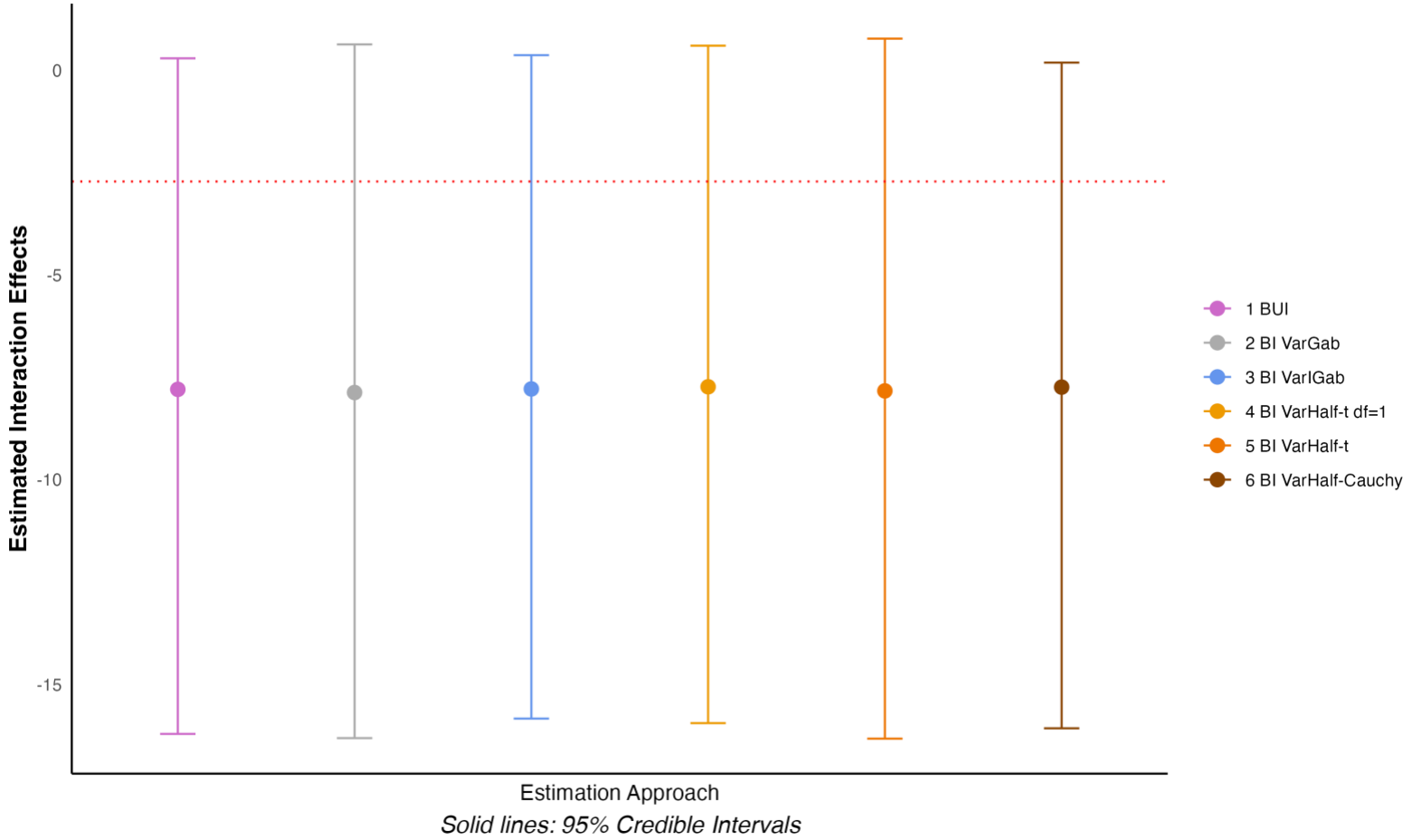
Estimating L2 Math (Aggregate, Z) Effects Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

Figure 36

Estimating Interaction Effects of L1 and L2 Math Using Informative Gamma, Inverse Gamma, Half-t, and Half-Cauchy (Subsample)



Note: BUI = Bayesian uninformative (flat) priors. BI = Bayesian with informative priors. VarGab = Gamma distribution *G*(a,b) priors for variance components; parameters a and *b* derived from chi-square distributions transformed to Gamma. VARIGab = Inverse Gamma *IG*(a,b) priors for variance components; involves converting chi-square values to Gamma, then to Inverse Gamma. Half-*t*(*df*, location, scale) used for variance component priors, with degree of freedom *df*, location, and scale parameters based on chi-square distribution (*df* = 1 or *df* = *k*). Half-Cauchy(0, scale) used for variance component priors, being a special case of the half-*t* distribution with one degree of freedom and a location parameter of zero.

APPENDIX

1. Example Mplus code for Simulation Data Generation (*J* = 10, *m* = 5, ICC = .40)

```
TITLE: J10_M5_ICC0.40_R2w0.50_R2b0.20_Pop;
MONTECARLO: NAMES ARE x1-x2 y1;
NOBSERVATIONS = 500;
NCSIZES = 1;
CSIZES = 100 (5);
NREPS = 200;
SEED = 293167;
RESULTS = J10_M5_ICC0.40_R2w0.50_R2b0.20_Pop.dat;
REPSAVE = all;
SAVE = J10_M5_ICC0.40_R2w0.50_R2b0.20_Pop_rep_*.dat;
ANALYSIS: TYPE=TWOLEVEL; ESTIMATOR = ML; PROCESSORS = 38;
MODEL POPULATION:
       %WITHIN%
       x1-x2@1;
       y1@0.500000;
       x1 with x2@0;
       y1 on x1-x2@0.500000;
       %BETWEEN%
       x1-x2@0.6666667;
       y1@0.5333333;
       x1 with x2@0;
       y1 on x1-x2@0.3162278;
MODEL:
  %WITHIN%
  y1 on x1-x2*0.500000;
  %BETWEEN%
  y1 on x1-x2*0.3162278;
```

2. Example Mplus code for Analyses with Uninformative Priors (*J* = 10, *m* = 5, ICC = .40)

```
TITLE: J10_M5_ICC0.40_R2w0.50_R2b0.20_Bui;
DATA:
FILE=C:\BMLM\Pop\J10_M5_ICC0.40_R2w0.50_R2b0.20_Pop_rep_list.dat;
  TYPE = MONTECARLO;
VARIABLE:
  NAMES = y1 x1-x2 Clust; USEVARIABLES = y1 x1-x2; CLUSTER = Clust;
ANALYSIS:
  TYPE=TWOLEVEL; ESTIMATOR = BAYES;
  MODEL PRIORS:
  p1 ~ IG(-1, 0);
  p2 ~ N(0, 10000);
  p3 ~ IG(-1, 0);
  p4 ~ N(0, 10000);
MODEL:
  %WITHIN%
  y1 (p1);
  y1 on x1-x2 (p2);
  %BETWEEN%
  y1 (p3);
  y1 on x1-x2 (p4) ;
```

3. Example Mplus code for Analyzing Datasets with Weak Informative Normal Coefficient Priors and Strong Informative Inverse Gamma Variance Priors (J = 10, m = 5, ICC = .40)

```
TITLE: J10_M5_ICC0.40_R2w0.50_R2b0.20_Bi_cv_IGab;
DATA:
FILE=C:\BMLM\Pop\J10_M5_ICC0.40_R2w0.50_R2b0.20_Pop_rep_list.dat;
  TYPE = MONTECARLO;
VARIABLE:
  NAMES = y1 x1-x2 Clust; USEVARIABLES = y1 x1-x2; CLUSTER = Clust;
ANALYSIS:
  TYPE=TWOLEVEL; ESTIMATOR = BAYES;
  MODEL PRIORS:
  p1 ~ IG(2.500000,0.750000);
  p2 ~ N(0.500000, 10000);
  p3 ~ IG(2.533333,0.8177778);
  p4 ~ N(0.3162278, 10000);
MODEL:
  %WITHIN%
  y1 (p1);
  y1 on x1-x2 (p2);
  %BETWEEN%
  y1 (p3);
  y1 on x1-x2 (p4);
```

4. Example R code for Real Data Analysis

```
# Import sav data into R
timss <- read_sav("bsausam7.sav")

# Subset data with only content domains for math and science ----
timss_wide_whole <- timss %>%
  select("IDCNTRY", "IDBOOK", "IDSCHOOL", "IDCLASS", "IDSTUD",
"BSMMAT01", "BSMMAT02", "BSMMAT03", "BSMMAT04", "BSMMAT05",
"BSSSCI01", "BSSSCI02", "BSSSCI03", "BSSSCI04", "BSSSCI05",
"BSMNUM01", "BSMNUM02", "BSMNUM03", "BSMNUM04", "BSMNUM05",
"BSMALG01", "BSMALG02", "BSMALG03", "BSMALG04", "BSMALG05",
"BSMGEO01", "BSMGEO02", "BSMGEO03", "BSMGEO04", "BSMGEO05",
"BSMDAT01", "BSMDAT02", "BSMDAT03", "BSMDAT04", "BSMDAT05",
"BSSBIO01", "BSSBIO02", "BSSBIO03", "BSSBIO04", "BSSBIO05",
"BSSCHE01", "BSSCHE02", "BSSCHE03", "BSSCHE04", "BSSCHE05",
"BSSPHY01", "BSSPHY02", "BSSPHY03", "BSSPHY04", "BSSPHY05",
"BSSEAR01", "BSSEAR02", "BSSEAR03", "BSSEAR04", "BSSEAR05")

# Full sample
timss_wide <- timss_wide_whole

# Create aggregates
### Aggregate L1 raw predictors into new L2 predictor dataset
L2agg = aggregate(x =
timss_wide[c("BSMMAT01","BSMMAT02","BSMMAT03","BSMMAT04","BSMMAT05",
"BSSSCI01","BSSSCI02","BSSSCI03","BSSSCI04","BSSSCI05")],
                  by = timss_wide_subsample[c("IDSCHOOL")], FUN = mean)
colnames(L2agg) =
c("IDSCHOOL","BSMMAT01_agg","BSMMAT02_agg","BSMMAT03_agg","BSMMAT04_agg",
"BSMMAT05_agg","BSSSCI01_agg","BSSSCI02_agg","BSSSCI03_agg","BSSSCI04_agg",
"BSSSCI05_agg")

### z-score aggregates ----
L2agg$ZBSMMAT01_agg <- as.numeric(scale(L2agg$BSMMAT01_agg, scale=TRUE))
L2agg$ZBSMMAT02_agg <- as.numeric(scale(L2agg$BSMMAT02_agg, scale=TRUE))
```

```
L2agg$ZBSMMAT03_agg <- as.numeric(scale(L2agg$BSMMAT03_agg, scale=TRUE))
L2agg$ZBSMMAT04_agg <- as.numeric(scale(L2agg$BSMMAT04_agg, scale=TRUE))
L2agg$ZBSMMAT05_agg <- as.numeric(scale(L2agg$BSMMAT05_agg, scale=TRUE))
L2agg$ZBSSSCI01_agg <- as.numeric(scale(L2agg$BSSSCI01_agg, scale=TRUE))
L2agg$ZBSSSCI02_agg <- as.numeric(scale(L2agg$BSSSCI02_agg, scale=TRUE))
L2agg$ZBSSSCI03_agg <- as.numeric(scale(L2agg$BSSSCI03_agg, scale=TRUE))
L2agg$ZBSSSCI04_agg <- as.numeric(scale(L2agg$BSSSCI04_agg, scale=TRUE))
L2agg$ZBSSSCI05_agg <- as.numeric(scale(L2agg$BSSSCI05_agg, scale=TRUE))

### merge aggregates
timss_wide <- merge(timss_wide, L2agg, by = "IDSCHOOL")

# Mean-center L1 math predictors using misty package
timss_wide$BSMMAT01_CMC <- as.numeric(center(timss_wide$BSMMAT01, type="CWC", cluster
=timss_wide$IDSCHOOL))
timss_wide$BSMMAT02_CMC <- as.numeric(center(timss_wide$BSMMAT02, type="CWC", cluster
=timss_wide$IDSCHOOL))
timss_wide$BSMMAT03_CMC <- as.numeric(center(timss_wide$BSMMAT03, type="CWC", cluster
=timss_wide$IDSCHOOL))
timss_wide$BSMMAT04_CMC <- as.numeric(center(timss_wide$BSMMAT04, type="CWC", cluster
=timss_wide$IDSCHOOL))
timss_wide$BSMMAT05_CMC <- as.numeric(center(timss_wide$BSMMAT05, type="CWC", cluster
=timss_wide$IDSCHOOL))
timss_wide$ZBSMMAT01_CMC = as.numeric(scale(timss_wide$BSMMAT01_CMC))
timss_wide$ZBSMMAT02_CMC = as.numeric(scale(timss_wide$BSMMAT02_CMC))
timss_wide$ZBSMMAT03_CMC = as.numeric(scale(timss_wide$BSMMAT03_CMC))
timss_wide$ZBSMMAT04_CMC = as.numeric(scale(timss_wide$BSMMAT04_CMC))
timss_wide$ZBSMMAT05_CMC = as.numeric(scale(timss_wide$BSMMAT05_CMC))

#Analysis with Full Sample
#0) ML estimation
ML_model01_ZBSMMAT01_start <- Sys.time()
ML_model01_ZBSMMAT01 <- lmer(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
ZBSMMAT01_CMC*ZBSMMAT01_agg +
                             (1 | IDSCHOOL),
                             data = timss_wide,
                             REML = FALSE)
ML_model01_ZBSMMAT01_end <- Sys.time()
ML_model01_ZBSMMAT01_tottime <- ML_model01_ZBSMMAT01_end - ML_model01_ZBSMMAT01_start
ML_model01_ZBSMMAT01_tottime
summary(ML_model01_ZBSMMAT01)

#Section 1 Bayesian estimation approaches for uninformative and informative inverse
#gamma priors

#1) BUI
##### Define the uniform priors for variance components
prior_fullsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd", group =
"IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")

##### Define the normal distribution priors for coefficients
prior_fullsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BUI <- c(prior_fullsample_L2_resid_sd_U,
prior_fullsample_L1_resid_sd_U, prior_fullsample_int_N,
```

```
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model11_ZBSMMAT01_start <- Sys.time()
bayes_model11_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BUI,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99)) # Adjust if necessary for model convergence

bayes_model11_ZBSMMAT01_end <- Sys.time()
bayes_model11_ZBSMMAT01_tottime <- bayes_model11_ZBSMMAT01_end -
bayes_model11_ZBSMMAT01_start
bayes_model11_ZBSMMAT01_tottime
posterior_summary(as.data.frame(bayes_model11_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#2) BI Coef N
##### Define the uniform priors for variance components
prior_fullsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd", group =
"IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_N <- c(prior_fullsample_L2_resid_sd_U,
prior_fullsample_L1_resid_sd_U, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model21_ZBSMMAT01_start <- Sys.time()
bayes_model21_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_N,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model21_ZBSMMAT01_end <- Sys.time()
bayes_model21_ZBSMMAT01_tottime <- bayes_model21_ZBSMMAT01_end -
bayes_model21_ZBSMMAT01_start
bayes_model21_ZBSMMAT01_tottime
```

```
posterior_summary(as.data.frame(bayes_model21_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#3) BI VarIG01
##### Define the inverse gamma priors for variance components
prior_fullsample_L2_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class =
"sigma")

##### Define the normal priors for coefficient priors
prior_fullsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000 )", class = "b",
coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000 )", class = "b",
coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(0,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_IG01 <- c(prior_fullsample_L2_resid_sd_IG01,
prior_fullsample_L1_resid_sd_IG01, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model31_ZBSMMAT01_start <- Sys.time()
bayes_model31_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
(1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_IG01, #remember to change
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model31_ZBSMMAT01_end <- Sys.time()
bayes_model31_ZBSMMAT01_tottime <- bayes_model31_ZBSMMAT01_end -
bayes_model31_ZBSMMAT01_start
bayes_model31_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model31_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#4) BI VarIGab
##### Define the inverse gamma priors for variance components
prior_fullsample_L2_resid_sd_IGab <- set_prior("inv_gamma(12, 154)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_IGab <- set_prior("inv_gamma(46, 2135)", class = "sigma")

##### Define the normal priors for coefficient priors
prior_fullsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC:ZBSMMAT01_agg")
```

```
priors_fullsample_BI_IGab <- c(prior_fullsample_L2_resid_sd_IGab,
prior_fullsample_L1_resid_sd_IGab, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)


bayes_model41_ZBSMMAT01_start <- Sys.time()
bayes_model41_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_IGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model41_ZBSMMAT01_end <- Sys.time()
bayes_model41_ZBSMMAT01_tottime <- bayes_model41_ZBSMMAT01_end -
bayes_model41_ZBSMMAT01_start
bayes_model41_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model41_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#5) BI Coef & VarIG01: Combine the weak fixed effect with weak random effect priors
(#2 and #3)
##### Define the inverse gamma priors for variance components
prior_fullsample_L2_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class =
"sigma")

##### Define the informative normal priors for coefficient
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_N_IG01 <- c(prior_fullsample_L2_resid_sd_IG01,
prior_fullsample_L1_resid_sd_IG01, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model51_ZBSMMAT01_start <- Sys.time()
bayes_model51_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_N_IG01,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
```

```
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model51_ZBSMMAT01_end <- Sys.time()
bayes_model51_ZBSMMAT01_tottime <- bayes_model51_ZBSMMAT01_end -
bayes_model51_ZBSMMAT01_start
bayes_model51_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model51_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#6) BI Coef & VarIGab: Combine the weak fixed effect with strong random effect priors
(#2 and #4)
##### Define the inverse gamma priors for variance components
prior_fullsample_L2_resid_sd_IGab <- set_prior("inv_gamma(12, 154)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_IGab <- set_prior("inv_gamma(46, 2135)", class = "sigma")

##### Define the informative normal priors for coefficient priors
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_N_IGab <- c(prior_fullsample_L2_resid_sd_IGab,
prior_fullsample_L1_resid_sd_IGab, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model61_ZBSMMAT01_start <- Sys.time()
bayes_model61_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_N_IGab, #remember to change
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model61_ZBSMMAT01_end <- Sys.time()
bayes_model61_ZBSMMAT01_tottime <- bayes_model61_ZBSMMAT01_end -
bayes_model61_ZBSMMAT01_start
bayes_model61_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model61_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#Section 2 Bayesian estimation approaches for uninformative, informative gamma,
#inverse gamma, half-t, and half-Cauchy priors

#1) BUI
##### Define the uniform priors for variance components
prior_fullsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd", group =
"IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")
```

```
##### Define the weak normal priors for coefficient priors
prior_fullsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b",
coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BUI <- c(prior_fullsample_L2_resid_sd_U,
prior_fullsample_L1_resid_sd_U, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model11_ZBSMMAT01_start <- Sys.time()
bayes_model11_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BUI,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99)) # Adjust if necessary for model convergence

bayes_model11_ZBSMMAT01_end <- Sys.time()
bayes_model11_ZBSMMAT01_tottime <- bayes_model11_ZBSMMAT01_end -
bayes_model11_ZBSMMAT01_start
bayes_model11_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model11_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#2) BI Gab
##### Define the informative gamma distribution for variance components
prior_fullsample_L2_resid_sd_VarGab <-  set_prior("gamma(9, 0.5)", class = "sd", group
= "IDSCHOOL",  coef = "Intercept")
prior_fullsample_L1_resid_sd_VarGab <- set_prior("gamma(33, 0.5)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BICoef_VarGab <- c(prior_fullsample_L2_resid_sd_VarGab,
prior_fullsample_L1_resid_sd_VarGab, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model21_ZBSMMAT01_start <- Sys.time()
bayes_model21_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
```

```
  family = gaussian(),
  prior = priors_fullsample_BICoef_VarGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model21_ZBSMMAT01_end <- Sys.time()
bayes_model21_ZBSMMAT01_tottime <- bayes_model21_ZBSMMAT01_end -
bayes_model21_ZBSMMAT01_start
bayes_model21_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model21_ZBSMMAT01),robust = TRUE) # uses median


#3) BI IGab
##### Define the informative inverse gamma distribution for variance components
prior_fullsample_L2_resid_sd_VarIGab <-  set_prior("inv_gamma(12, 154)", class = "sd",
group = "IDSCHOOL",  coef = "Intercept")
prior_fullsample_L1_resid_sd_VarIGab <- set_prior("inv_gamma(46, 2135)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BICoef_VarIGab <- c(prior_fullsample_L2_resid_sd_VarIGab,
prior_fullsample_L1_resid_sd_VarIGab, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model31_ZBSMMAT01_start <- Sys.time()
bayes_model31_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BICoef_VarIGab, #remember to change
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.999) # Higher target acceptance rate for NUTS
)
bayes_model31_ZBSMMAT01_end <- Sys.time()
bayes_model31_ZBSMMAT01_tottime <- bayes_model31_ZBSMMAT01_end -
bayes_model31_ZBSMMAT01_start
bayes_model31_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model31_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)


#4) BI Half-t df=1
##### Define the informative half-t distribution for variance components
```

```
prior_fullsample_L2_resid_sd_Half_t_df1 <-  set_prior("student_t(1, 12, 18)", class =
"sd", group = "IDSCHOOL",  coef = "Intercept")
prior_fullsample_L1_resid_sd_Half_t_df1 <- set_prior("student_t(1, 46, 65)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BICoef_Half_t_df1 <- c(prior_fullsample_L2_resid_sd_Half_t_df1,
prior_fullsample_L1_resid_sd_Half_t_df1, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model41_ZBSMMAT01_start <- Sys.time()
bayes_model41_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BICoef_Half_t_df1,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model41_ZBSMMAT01_end <- Sys.time()
bayes_model41_ZBSMMAT01_tottime <- bayes_model41_ZBSMMAT01_end -
bayes_model41_ZBSMMAT01_start
bayes_model41_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model41_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#5) BI Half-t with scale/loc and df computed from chi drawn from #1
##### Define the informative half-t distribution for variance components
prior_fullsample_L2_resid_sd_BI_Half_t <-  set_prior("student_t(272, 12, 18)", class =
"sd", group = "IDSCHOOL",  coef = "Intercept")
prior_fullsample_L1_resid_sd_BI_Half_t <- set_prior("student_t(8422, 46, 65)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_Half_t <- c(prior_fullsample_L2_resid_sd_BI_Half_t,
prior_fullsample_L1_resid_sd_BI_Half_t, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model51_ZBSMMAT01_start <- Sys.time()
```

```
bayes_model51_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
(1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_Half_t,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model51_ZBSMMAT01_end <- Sys.time()
bayes_model51_ZBSMMAT01_tottime <- bayes_model51_ZBSMMAT01_end -
bayes_model51_ZBSMMAT01_start
bayes_model51_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model51_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#6) BI Half-Cauchy with df = 1, location = 0, and keep scale the same as #5
##### Define the informative half-Cauchy distribution for variance components
prior_fullsample_L2_resid_sd_BI_Half_Cauchy <- set_prior("cauchy(0, 18)", class =
"sd", group = "IDSCHOOL", coef = "Intercept")
prior_fullsample_L1_resid_sd_BI_Half_Cauchy <- set_prior("cauchy(0, 65)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_fullsample_int_N <- set_prior("normal(523.22, 10000)", class = "Intercept")
prior_fullsample_slope_ZBSMMAT01_CMC <- set_prior("normal(59.13, 10000 )", class =
"b", coef = "ZBSMMAT01_CMC")
prior_fullsample_slope_ZBSMMAT01_agg <- set_prior("normal(64.20, 10000 )", class =
"b", coef = "ZBSMMAT01_agg")
prior_fullsample_slope_interaction <- set_prior("normal(-2.72,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_fullsample_BI_Half_Cauchy <- c(prior_fullsample_L2_resid_sd_BI_Half_Cauchy,
prior_fullsample_L1_resid_sd_BI_Half_Cauchy, prior_fullsample_int_N,
prior_fullsample_slope_ZBSMMAT01_CMC, prior_fullsample_slope_ZBSMMAT01_agg,
prior_fullsample_slope_interaction)

bayes_model61_ZBSMMAT01_start <- Sys.time()
bayes_model61_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
(1 | IDSCHOOL),
  data = timss_wide,
  family = gaussian(),
  prior = priors_fullsample_BI_Half_Cauchy,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_model61_ZBSMMAT01_end <- Sys.time()
bayes_model61_ZBSMMAT01_tottime <- bayes_model61_ZBSMMAT01_end -
bayes_model61_ZBSMMAT01_start
bayes_model61_ZBSMMAT01_tottime

posterior_summary(as.data.frame(bayes_model61_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)
```

```
#Analysis with Subsample
# Set the seed for reproducibility
set.seed(2949)

# Randomly select 30 schools
selected_schools <- timss_wide_whole %>%
  distinct(IDSCHOOL) %>%  # Ensure we are selecting from unique schools
  sample_n(30)  # Take a random sample of 30 schools

# Sample five students from each of the selected schools
timss_wide_subsample <- timss_wide_whole %>%
  filter(IDSCHOOL %in% selected_schools$IDSCHOOL) %>%
  group_by(IDSCHOOL) %>%  # Group data by school
  slice_sample(n = 5)  # Randomly select 5 students per school

# Create aggregates
### aggregate L1 raw predictors into new L2 predictor dataset----
L2agg = aggregate(x =
timss_wide_subsample[c("BSMMAT01","BSMMAT02","BSMMAT03","BSMMAT04","BSMMAT05",
"BSSSCI01","BSSSCI02","BSSSCI03","BSSSCI04","BSSSCI05")],
                  by = timss_wide_subsample[c("IDSCHOOL")], FUN = mean)
colnames(L2agg)c("IDSCHOOL","BSMMAT01_agg","BSMMAT02_agg","BSMMAT03_agg","BSMMAT04_agg
", "BSMMAT05_agg",
"BSSSCI01_agg","BSSSCI02_agg","BSSSCI03_agg","BSSSCI04_agg","BSSSCI05_agg")

### z-score aggregates
L2agg$ZBSMMAT01_agg <- as.numeric(scale(L2agg$BSMMAT01_agg, scale=TRUE))
L2agg$ZBSMMAT02_agg <- as.numeric(scale(L2agg$BSMMAT02_agg, scale=TRUE))
L2agg$ZBSMMAT03_agg <- as.numeric(scale(L2agg$BSMMAT03_agg, scale=TRUE))
L2agg$ZBSMMAT04_agg <- as.numeric(scale(L2agg$BSMMAT04_agg, scale=TRUE))
L2agg$ZBSMMAT05_agg <- as.numeric(scale(L2agg$BSMMAT05_agg, scale=TRUE))
L2agg$ZBSSSCI01_agg <- as.numeric(scale(L2agg$BSSSCI01_agg, scale=TRUE))
L2agg$ZBSSSCI02_agg <- as.numeric(scale(L2agg$BSSSCI02_agg, scale=TRUE))
L2agg$ZBSSSCI03_agg <- as.numeric(scale(L2agg$BSSSCI03_agg, scale=TRUE))
L2agg$ZBSSSCI04_agg <- as.numeric(scale(L2agg$BSSSCI04_agg, scale=TRUE))
L2agg$ZBSSSCI05_agg <- as.numeric(scale(L2agg$BSSSCI05_agg, scale=TRUE))

### merge aggregates
timss_wide_subsample <- merge(timss_wide_subsample, L2agg, by = "IDSCHOOL")


## Mean-center L1 math predictors using misty package
timss_wide_subsample$BSMMAT01_CMC <- as.numeric(center(timss_wide_subsample$BSMMAT01,
type="CWC", cluster =timss_wide_subsample$IDSCHOOL)) #### create z-scores of centered
variable
timss_wide_subsample$BSMMAT02_CMC <- as.numeric(center(timss_wide_subsample$BSMMAT02,
type="CWC", cluster =timss_wide_subsample$IDSCHOOL)) #### create z-scores of centered
variable
timss_wide_subsample$BSMMAT03_CMC <- as.numeric(center(timss_wide_subsample$BSMMAT03,
type="CWC", cluster =timss_wide_subsample$IDSCHOOL)) #### create z-scores of centered
variable
timss_wide_subsample$BSMMAT04_CMC <- as.numeric(center(timss_wide_subsample$BSMMAT04,
type="CWC", cluster =timss_wide_subsample$IDSCHOOL)) #### create z-scores of centered
variable
timss_wide_subsample$BSMMAT05_CMC <- as.numeric(center(timss_wide_subsample$BSMMAT05,
type="CWC", cluster =timss_wide_subsample$IDSCHOOL)) #### create z-scores of centered
variable
timss_wide_subsample$ZBSMMAT01_CMC =
as.numeric(scale(timss_wide_subsample$BSMMAT01_CMC))
timss_wide_subsample$ZBSMMAT02_CMC =
as.numeric(scale(timss_wide_subsample$BSMMAT02_CMC))
timss_wide_subsample$ZBSMMAT03_CMC =
as.numeric(scale(timss_wide_subsample$BSMMAT03_CMC))
```

```
timss_wide_subsample$ZBSMMAT04_CMC =
as.numeric(scale(timss_wide_subsample$BSMMAT04_CMC))
timss_wide_subsample$ZBSMMAT05_CMC =
as.numeric(scale(timss_wide_subsample$BSMMAT05_CMC))


#0) ML estimation
ML_subsample_model01_ZBSMMAT01_start <- Sys.time()
ML_subsample_model01_ZBSMMAT01 <- lmer(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
ZBSMMAT01_CMC*ZBSMMAT01_agg +
                            (1 | IDSCHOOL),
                            data = timss_wide_subsample,
                            REML = FALSE)
ML_subsample_model01_ZBSMMAT01_end <- Sys.time()
ML_subsample_model01_ZBSMMAT01_tottime <- ML_subsample_model01_ZBSMMAT01_end -
ML_subsample_model01_ZBSMMAT01_start
ML_subsample_model01_ZBSMMAT01_tottime
summary(ML_subsample_model01_ZBSMMAT01)


#Section 1 Bayesian estimation approaches for uninformative and informative inverse
#gamma priors

#1)BUI
##### Define the uniform priors for variance components
prior_subsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd", group =
"IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")

##### Define the normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_agg")
prior_subsample_slope_interaction   <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BUI <- c(prior_subsample_L2_resid_sd_U,
prior_subsample_L1_resid_sd_U, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model11_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model11_ZBSMMAT01 <- brm(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
  ZBSMMAT01_CMC*ZBSMMAT01_agg + (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BUI,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) #Adjust if necessary for model convergence
)
bayes_subsample_model11_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model11_ZBSMMAT01_tottime <- bayes_subsample_model11_ZBSMMAT01_end -
bayes_subsample_model11_ZBSMMAT01_start
bayes_subsample_model11_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model11_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)
```

```
#2) BI Coef N
##### Define the uniform priors for variance components
prior_subsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd", group =
"IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.22, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.71, 10000 )", class = "b",
coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.37, 10000 )", class = "b",
coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.80,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_N <- c(prior_subsample_L2_resid_sd_U,
prior_subsample_L1_resid_sd_U, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model21_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model21_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
  ZBSMMAT01_CMC * ZBSMMAT01_agg + (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_N,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)

bayes_subsample_model21_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model21_ZBSMMAT01_tottime <- bayes_subsample_model21_ZBSMMAT01_end -
bayes_subsample_model21_ZBSMMAT01_start
bayes_subsample_model21_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model21_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)


#3) BI VarIG01
##### Define the inverse gamma priors for variance components
prior_subsample_L2_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class =
"sigma")

##### Define the normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000 )", class = "b",
coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000 )", class = "b",
coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(0,  10000 )", class = "b", coef
= "ZBSMMAT01_CMC:ZBSMMAT01_agg")
```

```
priors_subsample_BI_IG01 <- c(prior_subsample_L2_resid_sd_IG01,
prior_subsample_L1_resid_sd_IG01, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model31_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model31_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_IG01,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model31_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model31_ZBSMMAT01_tottime <- bayes_subsample_model31_ZBSMMAT01_end -
bayes_subsample_model31_ZBSMMAT01_start
bayes_subsample_model31_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model31_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)


#4) BI VarIGab
##### Define the inverse gamma priors for variance components
prior_subsample_L2_resid_sd_IGab <- set_prior("inv_gamma(6, 34)", class = "sd", group
= "IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_IGab <- set_prior("inv_gamma(45, 2046)", class = "sigma")

##### Define the normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(0, 10000)", class = "b", coef =
"ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_IGab <- c(prior_subsample_L2_resid_sd_IGab,
prior_subsample_L1_resid_sd_IGab, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model41_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model41_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_IGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model41_ZBSMMAT01_end <- Sys.time()
```

```
bayes_subsample_model41_ZBSMMAT01_tottime <- bayes_subsample_model41_ZBSMMAT01_end -
bayes_subsample_model41_ZBSMMAT01_start
bayes_subsample_model41_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model41_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)


#5) BI Coef & VarIG01: Combine the weak fixed effect with weak random effect priors #2
and #3)
##### Define the inverse gamma priors for variance components
prior_subsample_L2_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_IG01 <- set_prior("inv_gamma(0.01, 0.01)", class =
"sigma")

##### Define the informative normal priors for coefficient priors
prior_subsample_int_N <- set_prior("normal(519.22, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.71, 10000 )", class = "b",
coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.37, 10000 )", class = "b",
coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.80,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_N_IG01 <- c(prior_subsample_L2_resid_sd_IG01,
prior_subsample_L1_resid_sd_IG01, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model51_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model51_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_N_IG01,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model51_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model51_ZBSMMAT01_tottime <- bayes_subsample_model51_ZBSMMAT01_end -
bayes_subsample_model51_ZBSMMAT01_start
bayes_subsample_model51_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model51_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)


#6) BI Coef & VarIGab: Combine the weak fixed effect with strong random effect priors
(#2 and #4)
##### Define the inverse gamma priors for variance components
prior_subsample_L2_resid_sd_IGab <- set_prior("inv_gamma(6, 34)", class = "sd", group
= "IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_IGab <- set_prior("inv_gamma(45, 2046)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.22, 10000)", class = "Intercept")
```

```
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.71, 10000 )", class = "b",
coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.37, 10000 )", class = "b",
coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.80,  10000 )", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_N_IGab <- c(prior_subsample_L2_resid_sd_IGab,
prior_subsample_L1_resid_sd_IGab, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model61_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model61_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_N_IGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model61_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model61_ZBSMMAT01_tottime <- bayes_subsample_model61_ZBSMMAT01_end -
bayes_subsample_model61_ZBSMMAT01_start
bayes_subsample_model61_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model61_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)


#Section 2 Bayesian estimation approaches for uninformative, informative gamma,
#inverse gamma, half-t, and half-Cauchy.

#1) BUI
##### Define the uniform priors for variance components
prior_subsample_L2_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sd",
                                           group = "IDSCHOOL", coef = "Intercept")
prior_subsample_L1_resid_sd_U <- set_prior("uniform(0, 10000)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(0, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(0, 10000)", class = "b", coef
= "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(0, 10000)", class = "b", coef =
"ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BUI <- c(prior_subsample_L2_resid_sd_U,
prior_subsample_L1_resid_sd_U, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model11_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model11_ZBSMMAT01 <- brm(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
ZBSMMAT01_CMC*ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
```

```
  family = gaussian(),
  prior = priors_subsample_BUI,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) #Adjust if necessary for model convergence
)
bayes_subsample_model11_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model11_ZBSMMAT01_tottime <- bayes_subsample_model11_ZBSMMAT01_end -
bayes_subsample_model11_ZBSMMAT01_start
bayes_subsample_model11_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model11_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)

#### Save BUI values for subsequent informative priors
prior_coef_Intcpt <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$b_
Intercept,robust = TRUE)[[1]]),4)
prior_coef_L1pred <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$b_
ZBSMMAT01_CMC,robust = TRUE)[[1]]),4)
prior_coef_L2pred <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$b_
ZBSMMAT01_agg,robust = TRUE)[[1]]),4)
prior_coef_L1xL2int <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$"b
ZBSMMAT01_CMC:ZBSMMAT01_agg",robust = TRUE)[[1]]),4)
prior_L2reserrSD <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$sd
_IDSCHOOL__Intercept,robust = TRUE)[[1]]),4)
prior_L1reserrSD <-
round(as.numeric(posterior_summary(as.data.frame(bayes_subsample_model11_ZBSMMAT01)$si
gma,robust = TRUE)[[1]]),4)

#2) BI Gab
##### Define the gamma priors for variance components
prior_subsample_L2_resid_sd_VarGab <-  set_prior("gamma(4, 0.5)", class = "sd", group
= "IDSCHOOL",  coef = "Intercept")
prior_subsample_L1_resid_sd_VarGab <- set_prior("gamma(32, 0.5)", class = "sigma")

##### Define the informative normal priors for coefficients ----
prior_subsample_int_N <- set_prior("normal(519.2215, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.7128, 10000)", class =
"b", coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.3653, 10000)", class =
"b", coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.7958, 10000)", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BICoef_VarGab <- c(prior_subsample_L2_resid_sd_VarGab,
prior_subsample_L1_resid_sd_VarGab, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model21_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model21_ZBSMMAT01 <- brm(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
ZBSMMAT01_CMC*ZBSMMAT01_agg +
 (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
```

```
  prior = priors_subsample_BICoef_VarGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model21_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model21_ZBSMMAT01_tottime <- bayes_subsample_model21_ZBSMMAT01_end -
bayes_subsample_model21_ZBSMMAT01_start
bayes_subsample_model21_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model21_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)

#3) BI IGab
##### Define the inverse gamma priors for variance components
prior_subsample_L2_resid_sd_VarIGab <- set_prior("inv_gamma(6, 34)", class = "sd",
group = "IDSCHOOL",  coef = "Intercept")
prior_subsample_L1_resid_sd_VarIGab <- set_prior("inv_gamma(45, 2046)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.2215, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal( 64.7128, 10000)", class =
"b", coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal( 72.3653, 10000)", class =
"b", coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal( -7.7958, 10000)", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BICoef_VarIGab <- c(prior_subsample_L2_resid_sd_VarIGab,
prior_subsample_L1_resid_sd_VarIGab, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model31_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model31_ZBSMMAT01 <- brm(BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg +
ZBSMMAT01_CMC*ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BICoef_VarIGab,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model31_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model31_ZBSMMAT01_tottime <- bayes_subsample_model31_ZBSMMAT01_end -
bayes_subsample_model31_ZBSMMAT01_start
bayes_subsample_model31_ZBSMMAT01_tottime

posterior_summary(bayes_subsample_model31_ZBSMMAT01,robust = TRUE)
# median absolute deviation (MAD)

#4)Bayes weak normal informative with PE from #1 and Half-t df=1 with scale/loc from
##### Define the half-t priors for variance components
prior_subsample_L2_resid_sd_Half_t_df1 <-  set_prior("student_t(1, 5.75, 8.3741)",
class = "sd", group = "IDSCHOOL",  coef = "Intercept")
```

```
prior_subsample_L1_resid_sd_Half_t_df1 <- set_prior("student_t(1, 45.22,
63.98200372)", class = "sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.2215, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal( 64.7128, 10000)", class =
"b", coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal( 72.3653, 10000)", class =
"b", coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal( -7.7958, 10000)", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BICoef_Half_t_df1 <- c(prior_subsample_L2_resid_sd_Half_t_df1,
prior_subsample_L1_resid_sd_Half_t_df1, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)
bayes_subsample_model41_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model41_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BICoef_Half_t_df1,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model41_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model41_ZBSMMAT01_tottime <- bayes_subsample_model41_ZBSMMAT01_end -
bayes_subsample_model41_ZBSMMAT01_start
bayes_subsample_model41_ZBSMMAT01_tottime

> posterior_summary(as.data.frame(bayes_subsample_model41_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#5) Bayes weak normal informative with PE from #1 and Half-t with scale/loc and df
computed from chi drawn from #1
##### Define the half-t priors for variance components
prior_subsample_L2_resid_sd_BI_Half_t <-  set_prior("student_t(29, 6, 8)", class =
"sd", group = "IDSCHOOL",  coef = "Intercept")
prior_subsample_L1_resid_sd_BI_Half_t <- set_prior("student_t(117, 45, 64)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.2215, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.7128, 10000)", class =
"b", coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.3653, 10000)", class =
"b", coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.7958, 10000)", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_Half_t <- c(prior_subsample_L2_resid_sd_BI_Half_t,
prior_subsample_L1_resid_sd_BI_Half_t, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model51_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model51_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
```

```
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_subsample_BI_Half_t,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model51_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model51_ZBSMMAT01_tottime <- bayes_subsample_model51_ZBSMMAT01_end -
bayes_subsample_model51_ZBSMMAT01_start
bayes_subsample_model51_ZBSMMAT01_tottime #25.45786 secs
posterior_summary(as.data.frame(bayes_subsample_model51_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)

#6) Bayes weak normal informative with PE from #1 and Half-Cauchy with df = 1,
location = 0, and keep scale the same as #5
##### Define the half-Cauchy priors for variance components
prior_subsample_L2_resid_sd_BI_Half_Cauchy <- set_prior("cauchy(0, 8)", class = "sd",
group = "IDSCHOOL", coef = "Intercept")

prior_subsample_L1_resid_sd_BI_Half_Cauchy <- set_prior("cauchy(0, 64)", class =
"sigma")

##### Define the informative normal priors for coefficients
prior_subsample_int_N <- set_prior("normal(519.2215, 10000)", class = "Intercept")
prior_subsample_slope_ZBSMMAT01_CMC <- set_prior("normal(64.7128, 10000)", class =
"b", coef = "ZBSMMAT01_CMC")
prior_subsample_slope_ZBSMMAT01_agg <- set_prior("normal(72.3653, 10000)", class =
"b", coef = "ZBSMMAT01_agg")
prior_subsample_slope_interaction <- set_prior("normal(-7.7958, 10000)", class = "b",
coef = "ZBSMMAT01_CMC:ZBSMMAT01_agg")

priors_subsample_BI_Half_Cauchy <- c(prior_subsample_L2_resid_sd_BI_Half_Cauchy,
prior_subsample_L1_resid_sd_BI_Half_Cauchy, prior_subsample_int_N,
prior_subsample_slope_ZBSMMAT01_CMC, prior_subsample_slope_ZBSMMAT01_agg,
prior_subsample_slope_interaction)

bayes_subsample_model61_ZBSMMAT01_start <- Sys.time()
bayes_subsample_model61_ZBSMMAT01 <- brm(
  formula = BSSSCI01 ~ ZBSMMAT01_CMC + ZBSMMAT01_agg + ZBSMMAT01_CMC * ZBSMMAT01_agg +
  (1 | IDSCHOOL),
  data = timss_wide_subsample,
  family = gaussian(),
  prior = priors_fullsample_BI_Half_Cauchy,
  seed = 2949,
  chains = 4,     # Default number of chains
  iter = 2000,    # Default total number of iterations per chain
  warmup = 1000,  # Default number of warmup/burn-in iterations
  thin = 1,       # Default thinning rate
  control = list(adapt_delta = 0.99) # Higher target acceptance rate for NUTS
)
bayes_subsample_model61_ZBSMMAT01_end <- Sys.time()
bayes_subsample_model61_ZBSMMAT01_tottime <- bayes_subsample_model61_ZBSMMAT01_end -
bayes_subsample_model61_ZBSMMAT01_start
bayes_subsample_model61_ZBSMMAT01_tottime
posterior_summary(as.data.frame(bayes_subsample_model61_ZBSMMAT01),robust = TRUE)
# median absolute deviation (MAD)
```