The variable importance metric LMG applied to Bayesian linear regression models

---

Master Thesis in Biostatistics (STA495)

by

Silvano Sele
09-734-492

supervised by

Dr. Stefanie Muff
Department of Biostatistics
Epidemiology, Biostatistics and Prevention Institute
University of Zurich

Zurich, September, 2018

Abstract

Quantifying the importance of predictors in regression has been an active area of research for a long time (Grömping, 2015). The variance decomposition metric LMG (named after the authors Lindeman, Merenda, and Gold) provides useful information about possible associations between variables (Grömping, 2007). The LMG metric is implemented in the R packages `hier.part` (Walsh and Nally, 2015) and `relaimpo` (Grömping, 2006) for models fitted in the frequentist framework. Bayesian methods gained high popularity in many applied research areas in recent years.

This master thesis shows how the LMG metric can be applied to a linear regression model that is fitted in the Bayesian framework. The LMG metric requires calculation of R^2 for all possible combinations of predictors. The conditional variance formula can be applied to calculate the R^2 values of these models containing only a subset of the predictors (sub-models) from the posterior samples of the model containing all predictors. The mutual interdependence of the sub-models is then respected for each posterior sample. The implementation of the LMG metric in the Bayesian framework is presented on simulated and on empirical data. Using weakly informative priors resulted in very similar LMG values as the values obtained by using bootstrap in `relaimpo`.

There are certain difficulties involved in quantifying the R^2 in longitudinal data. In this master thesis, some possible extensions of the LMG formula for the simple random intercept model as well as for marginal models, where the covariance structure of the error term is modeled, are sketched.

Contents

Chapter 1

Introduction

The objective of this master thesis is to implement the variable importance measure LMG (named after the authors Lindeman, Merenda, and Gold (Grömping, 2007)) in linear models estimated with Bayesian methods. Bayesian methods have gained popularity because they allow to quantify the uncertainty about parameters and they allow to include prior information.

Regression models are popular in many applied research areas (e.g. Nimon and Oswald, 2013). These models provide a tool to find an association between a response variable $\boldsymbol{y}$ and a set of explanatory variables $\mathbf{X}$ (Jackman, 2009). The explanatory variables are also called predictors or covariates. The conditional mean of a continuous response variable $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ is related to a $n \times k$ predictor matrix $\mathbf{X}$ via a linear model,

$$\mathrm{E}(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}) = \mathbf{X}\boldsymbol{\beta},$$

where $\boldsymbol{\beta}$ is a $k \times 1$ vector of unknown regression coefficients.

Under some assumptions about the density, conditional independence, and homoscedastic variances, the regression setting can be written as

$$\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2 \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n).$$

Regression parameters provide information to what extent the response variable is expected to change when one predictor changes by one unit, given all other predictors in the model remain the same. Being aware of this last remark is very important for the correct interpretation of the regression parameters, because it implies that the parameter value of a predictor is dependent on the other predictors in the model.

Because predictors are often correlated in real-world data to some degree to each other, it is obviously not an easy task to find the most important predictors in a model. The first question is what is meant by the importance of a predictor. There is no easy answer to this question and it is depending on the research issue. Grömping (2015) concludes that there may never be a unique definition of variable importance when predictors are correlated. There exist different metrics to quantify the importance of predictors. These metrics focus on different aspects and with correlated predictors they may lead to different conclusions. An overview of variable importance metrics can be found in Grömping (2015).

A distinction should be made between the importance of predictors in regression models that are used to predict future data and in regression models applied to find an association between predictors and the response variable. In the first case, the aim is only to reduce the error between the predicted values and the observable values. The underlying association between predictors is of minor importance. In the second case, the focus is on the strength of the relationship between the predictors and the response variable. A predictor may explain little of the response variable, given two other correlated predictors are already included in a regression model. However, this predictor, that is unimportant from the regression output, may be the main cause of the other

two predictors. Therefore, it may be the most important predictor of this regression model (Grömping, 2007).

The causal relationship between the variables is missing in standard regression models. Studying a predictor, given other variables are already included or using models that contain only the predictor itself, provide only some parts of the bigger picture about the predictor in a model. Which are the most useful variable importance metrics is still an open debate. A convincing theoretical basis is still lacking for all of those metrics. Grömping (2015) recommends to use the existing best practices until a more profound solution will be found. For variance (or generally goodness-of-fit) decomposition based importance, Grömping (2015) recommends to use LMG enhanced with joint contributions or dominance analysis.

The LMG metric is implemented for models fitted in the frequentist framework in the R packages `hier.part` (Walsh and Nally, 2015) and `relaimpo` (Grömping, 2006). Bayesian methods have gained high popularity in many applied sciences in recent years (e.g. van de Schoot *et al.*, 2017). A major advantage of Bayesian inference is that the posterior distributions of the parameters and the transformations of these parameters (like R^2 and the LMG metric) can be obtained. These posterior distributions allow to quantify the uncertainty about parameters and their transformations in an easily interpretable manner. This master thesis shows how the LMG metric can be applied to linear regression models fitted in the Bayesian framework. The LMG metric requires calculation of R^2 for all possible subsets of predictors. It shows that the R^2 of the models containing only a subset of predictors (sub-model) can be calculated from the model containing all predictors (full-model). Although this master thesis focuses on the LMG metric, the same approach could be used for other variance decomposition metrics that are based on the R^2 of the full-model and on the R^2 of the sub-models (e.g. commonality analysis (Nimon *et al.*, 2008) or dominance analysis (Grömping, 2015)).

The necessary background information is provided in chapter 2. Afterwards, the implementation of the LMG metric applied to Bayesian regression models is presented on simulated and on empirical data in chapter 3. Some difficulties and possible extensions of the LMG metric to longitudinal data are discussed in chapter 4. A general conclusion can be found in chapter 5.

Chapter 2

Theory

2.1 LMG variable importance metric

The LMG is a metric that is based on variance decomposition. The total R^2 of a model is decomposed onto the predictors. Marginal information (the association between a predictor and the dependent variable) and conditional information (the association of a predictor and the dependent variable given other predictors are already included) are incorporated (Grömping, 2015). The formulas in this section are taken from Grömping (2015), using the same mathematical notation.

The following notation for the explained variance (2.1) and sequentially added variance (2.2) of the predictors simplifies the notation of the LMG formula:

$$\text{evar}(S) = \text{Var}(Y) - \text{Var}(Y \mid X_j, j \in S), \tag{2.1}$$

$$\text{svar}(M \mid S) = \text{evar}(M \cup S) - \text{evar}(S), \tag{2.2}$$

where S and M denote disjoint sets of the predictor indices and X_j represents the set of predictors with indices from S. $R^2(S)$ can be written as $\text{evar}(S)/\text{Var}(Y)$ (Grömping, 2015).

The LMG formula is given below for the first predictor only. Because of exchangeable predictors, this is no loss of generality.

$$\begin{aligned}
\text{LMG}(1) &= \frac{1}{p!} \sum_{\pi permutation} \text{svar}(\{1\} \mid S_1(\pi)), && (2.3)\\
&= \frac{1}{p!} \sum_{S \subseteq \{2,\ldots,p\}} n(S)!\,(p - n(S) - 1)!\,\text{svar}(\{1\} \mid S) && (2.4)\\
&= \frac{1}{p} \sum_{i=0}^{p-1} \left(\sum_{\substack{S \subseteq \{2,\ldots,p\} \\ n(S)=i}} \text{svar}(\{1\} \mid S) \right) \Bigg/ \binom{p-1}{i} && (2.5)
\end{aligned}$$

where $S_1(\pi)$ is the set of predecessors of predictor 1 in permutation π (Grömping, 2015).

The different formula writings help to better understand what the calculation is about in the LMG metric. The R^2 of the model including all predictors is decomposed. In the formula on the top (2.3), the LMG value of predictor 1 is represented as an unweighted average over all orderings of the sequential added variance contribution of predictor 1. The formula in the center (2.4) shows that the calculation can be done more efficiently. The orderings with the same set of predecessors S are combined into one summand. Instead of $p!$ summands, only 2^{p-1} summands need to be calculated (Grömping, 2007). The formula on the bottom (2.5) shows that the LMG metric can also be seen as the unweighted average over the average explained variance

improvements when adding predictor 1 to a model of size i without predictor 1 (Grömping, 2015). The LMG metric is implemented in the R package `relaimpo` (Grömping, 2006).

Several authors formulated requirements that a variable importance metric should fulfill (Grömping, 2015). The following listed requirements are the two most important ones for variance decomposition metrics. The complete collection can be found in Grömping (2015).

a *Proper decomposition of the model variance:* the sum of all shares is the model variance (or R^2, depending on normalization). This is the defining criterion for variance decomposition metrics.

b *Non-negativity:* all allocated shares are always non-negative. This criterion is requested by many authors for variance decomposition metrics.

The LMG metric fulfills both requirements in the frequentist setting. There are some difficulties involved with the non-negativity property in the Bayesian framework that are considered in section 2.3.

Chevan and Sutherland (1991) propose that, instead of only using R^2, an appropriate goodness-of-fit metric can as well be used in the LMG formula. They name their proposal *hierarchical partitioning*. The requirements are simply: an initial measure of fit when no predictor variable is present, a final measure of fit when p predictor variables are present, and a measure of fit of all sub-models when various combinations of predictor variables are present. When R^2 is chosen as the goodness-of-fit measure, the standard LMG values are calculated. The LMG value of each variable is named *independent component* (I). The sum of the independent components (I) results in the overall goodness-of-fit metric of the model. The difference between the goodness-of-fit value of a model containing only the predictor itself to the value of its independent component (I) is named the *joint contribution* (J) (Grömping, 2015). As an example, a model containing only predictor $\mathbf{X}_1$ results in an R^2 of 0.3 and an LMG value of 0.2 (I). The joint contribution (J) is obtained by calculating $0.3 - 0.2 = 0.1$. Hierarchical partitioning is implemented in the `hier.part` package (Walsh and Nally, 2015). This function of `hier.part` is used in this master thesis. The hierarchical partitioning function accepts a data frame with the R^2 values of all sub-models as input. Of note, the partitioning function of `hier.part` is only guaranteed to work for up to nine predictors and it does not work at all for more than twelve predictors.

2.2 Appropriate R^2 definitions in the Bayesian framework

The focus of this master thesis is on the standard linear model. The most widely used goodness-of-fit metric for this model is R^2. There exist different formulas for R^2 (Kvalseth, 1985), all leading to the same value when an intercept is included and the model is fitted by maximum likelihood.

Two commonly used R^2 definitions are:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.6}$$

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \qquad i = 1, \ldots, n, \tag{2.7}$$

where y_i are the observations with indexes $i = 1, \ldots, n$, of sample size n, $\bar{y}$ represents the mean of the observations, $\hat{y}_i = \mathrm{E}(y_i \mid X_i, \hat{\theta})$, and $\hat{\theta}$ is the maximum likelihood estimate of the regression coefficients.

When other estimation methods than maximum likelihood are used, definition (2.6) can be < 0 and definition (2.7) can be > 1. This is not uncommon in a Bayesian regression setting

when samples of the posterior parameter distribution are employed (Gelman *et al.*, 2017). A model that explains more than 100% of the variance is nonsense. A negative R^2 is also difficult to interpret, although it may imply that the fit is worse than the mean of the observed sample. This can make sense for predictive purposes, e.g. when data from a test set is predicted by leave-one-out crossvalidation (Alexander *et al.*, 2015). It is then possible that the predicted values are in total further away (squared distance) from the test set observations than the mean of the test set observations. For non-predicting purposes, a negative R^2 does not make sense. The aim of the LMG formula is to gain some more information about the possible association between variables, and a predictor cannot explain less than zero variance in the population. To respect the non-negative share property of the LMG formula, the R^2 of sub-models should not decrease when adding predictors. Therefore, both classical R^2 definitions seem not to be well suited for the LMG metric in the Bayesian framework.

A useful R^2 definition for the LMG formula in the Bayesian framework can be found by noting that the variance of the linear model can also be written as

$$\text{Var}(y) = \text{Var}(\mathbf{X}\boldsymbol{\beta}) + \sigma^2 = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta} + \sigma^2, \tag{2.8}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ represents the regression parameters without the intercept of size $p \times 1$, $\mathbf{X}$ represents the predictor matrix of size $n \times p$ with corresponding covariance matrix $\boldsymbol{\Sigma}_{\mathbf{XX}}$ of size $p \times p$, and σ^2 represents the variance parameter.

In the Bayesian setting Gelman *et al.* (2017) propose to use

$$R_s^2 = \frac{\sum_{i=1}^n (\hat{y}_i^s - \bar{y^s})^2}{\sum_{i=1}^n (\hat{y}_i^s - \bar{y^s})^2 + \sum_{i=1}^n (e_i^s - \bar{e^s})^2}, \qquad i = 1, \ldots, n, \tag{2.9}$$

where $\hat{y}_i^s = \text{E}(y \mid X_i, \boldsymbol{\beta}^s)$ with corresponding mean $\bar{y^s}$, the errors $e_i^s = y_i - \hat{y}_i^s$ with corresponding mean $\bar{e^s}$, $\boldsymbol{\beta}^s$ represents the vector of regression parameters of size $p \times 1$ of draws, $s = 1, \ldots, S$, from the joint posterior parameter distribution. The R^2 is then guaranteed to be between 0 and 1. It can be interpreted as a data-based estimate of the proportion of variance explained for new data under the assumption that the predictors are held fixed (Gelman *et al.*, 2017).

In the Bayesian framework, the σ^2 parameter is explicitly modeled in the standard linear regression setting. Therefore, it is possible to sample the σ^2 parameter from its posterior distribution instead of defining the error as in definition (2.9), which would lead to the following definition:

$$\begin{aligned} R_s^2 &= \frac{\sum_{i=1}^n (\hat{y}_i^s - \bar{y^s})^2}{\sum_{i=1}^n (\hat{y}_i^s - \bar{y^s})^2 + \sigma_s^2} \\ &= \frac{\boldsymbol{\beta}_s^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}_s}{\boldsymbol{\beta}_s^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}_s + \sigma_s^2}, \qquad i = 1, \ldots, n, \end{aligned} \tag{2.10}$$

where the same notation as in (2.9) is used for the explained variance term and σ_s^2 represents the variance of the error term of draws, $s = 1, \ldots, S$, sampled from the joint posterior distribution.

In practice, definition (2.10) and definition (2.9) should lead to similar values in the standard linear model. In my opinion, it is more reasonable to take the full Bayesian route by sampling σ^2 of its posterior distribution. This approach provides the opportunity to include prior information about σ^2. The LMG calculations of the examples in this master thesis are therefore based on definition (2.10). However, a benefit of definition (2.9) is that it also works for generalized linear models where there is often no separate variance parameter.

2.3 Use of conditional variance formula to obtain R^2 of sub-models

The denominator of R^2 is no longer fixed in definition (2.9) and in definition (2.10). We can therefore no longer interpret an increase in R^2 as an improved fit to a fixed target (Gelman *et al.*,

2017). The unfixed denominator seems to be problematic for the LMG formula in the Bayesian framework. However, it is possible in the linear model to calculate the R^2 of all sub-models from the parameters of the full-model and the covariance matrix of the predictors. All sub-models of a posterior sample are then compared to the same fixed total variance value of a posterior sample (denominator in (2.10)). The mutual interdependence of the sub-models and the important non-negativity criterion is then respected for each posterior sample. How it is possible to obtain the R^2 of the sub-models from the full-model is shown in the following section.

Before the general case with p regressors is presented, a simple model with only two predictors is considered. For two predictors, $\mathbf{X}_{1,2} = (X_1, X_2)$, the covariance matrix can be written as

$$\operatorname{Cov}(\mathbf{X}_{1,2}) = \begin{pmatrix} \operatorname{Var}(X_1) & \operatorname{Cov}(X_1, X_2) \\ \operatorname{Cov}(X_1, X_2) & \operatorname{Var}(X_2) \end{pmatrix},$$

Definition (2.8) simplifies then to

$$\operatorname{Var}(y) = \beta_1^2 \operatorname{Var}(X_1) + 2\beta_1\beta_2 \operatorname{Cov}(X_1, X_2) + \beta_2^2 \operatorname{Var}(X_2) + \sigma^2. \tag{2.11}$$

It can be shown that if only X_1 is included in the model, the explained variance includes the variance of the predictor itself, the whole covariance term, and additionally some of the contribution of the variance of X_2 in equation (2.11). In mathematical notation, that is

$$\operatorname{evar}(X_1) = \beta_1^2 \operatorname{Var}(X_1) + 2\beta_1\beta_2 \operatorname{Cov}(X_1, X_2) + \beta_2^2 \operatorname{Var}(X_2)\rho_{12}^2, \tag{2.12}$$

where

$$\rho_{12} = \frac{\operatorname{Cov}(X_1, X_2)}{\sqrt{\operatorname{Var}(X_1)}\sqrt{\operatorname{Var}(X_2)}}$$

represents the correlation between predictors X_1 and X_2. The contribution of the second regressor is then simply the difference to the total explained variance (Grömping, 2007).

An alternative to obtain equation (2.12) is to subtract the conditional variance of predictor X_2 given predictor X_1 from the total explained variance of the full-model, in mathematical notation that is

$$\operatorname{evar}(X_1) = \beta_1^2 \operatorname{Var}(X_1) + 2\beta_1\beta_2 \operatorname{Cov}(X_1, X_2) + \beta_2^2 \operatorname{Var}(X_2) - \beta_2 \operatorname{Var}(X_2 \mid x_1)\beta_2, \tag{2.13}$$

where

$$\operatorname{Var}(X_2 \mid x_1) = \operatorname{Var}(X_2) - \frac{\operatorname{Cov}(X_1, X_2)^2}{\operatorname{Var}(X_1)}.$$

Equation (2.13) can easily be implemented for the general case with p regressors. The aim is to calculate R^2 of a sub-model containing the predictors $\mathbf{X}_{q,\cdots,p}$ and the regression coefficients $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ without the intercept of size $p \times 1$. The regression coefficients are further separated in $\boldsymbol{\beta}_{1,\ldots,q-1} = (\beta_1, \ldots, \beta_{q-1})$ and $\boldsymbol{\beta}_{q,\ldots,p} = (\beta_q, \ldots, \beta_p)$.

The covariance matrix of p predictors is written as

$$\operatorname{Cov}(\mathbf{X}) = \boldsymbol{\Sigma}_{\mathbf{XX}} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}^{p \times p},$$

$$\begin{aligned} \text{where} \quad & \boldsymbol{\Sigma}_{11} = \operatorname{Cov}(\mathbf{X}_{1,\ldots,q-1}, \mathbf{X}_{1,\ldots,q-1}), \\ & \boldsymbol{\Sigma}_{12} = \operatorname{Cov}(\mathbf{X}_{1,\ldots,q-1}, \mathbf{X}_{q,\ldots,p}) = \boldsymbol{\Sigma}_{21}^{\top}, \\ & \boldsymbol{\Sigma}_{22} = \operatorname{Cov}(\mathbf{X}_{q,\ldots,p}, \mathbf{X}_{q,\ldots,p}). \end{aligned}$$

The conditional variance of the predictors $\mathbf{X}_{1,\ldots,q-1}$, given the predictors $\mathbf{X}_{q,\ldots,p}$, is then

$$\text{Cov}(\mathbf{X}_{1,\ldots,q-1} \mid \boldsymbol{x}_{q,\ldots,p}) = \boldsymbol{\Sigma}_{11} \; - \; \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}. \tag{2.14}$$

The total explained variance of the full-model containing $\mathbf{X}_{1,\ldots,p}$ omits simply the σ^2 parameter in equation (2.8) , which is

$$\text{evar}(\mathbf{X}_{1,\ldots,p}) = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}.$$

The explained variance of a sub-model can be calculated by subtracting the explained variance of the not-in-the-model-included predictors that is not explained by in-the-model-included predictors from the total explained variance. The variance that is not explained by in-the-model-included predictors is given by the variance of the not-in-the-model-included predictors conditional on the in-the-model-included predictors. The explained variance of a sub-model containing predictors $\mathbf{X}_{q,\ldots,p}$ can therefore be written as

$$\text{evar}(\mathbf{X}_{q,\ldots,p}) = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta} - \boldsymbol{\beta}_{1,\ldots,q-1}^\top \text{Cov}(\mathbf{X}_{1,\ldots,q-1} \mid \boldsymbol{x}_{q,\ldots,p}) \boldsymbol{\beta}_{1,\ldots,q-1}. \tag{2.15}$$

To gain the the R^2 value of the sub-model, it is necessary to divide the explained variance by the total variance, which is

$$\text{evar}(\mathbf{X}_{q,\ldots,p}) / \text{Var}(y),$$

where $\text{Var}(y)$ is defined as $\boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta} + \sigma^2$.

A posterior density distribution is obtained for the regression parameters in the Bayesian regression setting. The LMG formula requires calculation of the R^2 values for all $2^p - 1$ sub-models. Samples from the joint posterior parameters of the full-model are used to calculate the explained variance of the sub-models. For each sample, the conditional variance formula is used to obtain the R^2 of the $2^p - 1$ sub-models. The non-negative property and the dependence of the parameters from the sub-models to each other is then respected for each sample.

Instead of using the conditional variance formula (2.14) to get the R^2 of the sub-models, it would have been possible to fit a separate Bayesian model for each sub-model. An R^2 distribution can easily be built for each sub-model by using definition (2.9) or definition (2.10). However, the problem is how to calculate the LMG values out of these R^2 distributions. If we just sample independently from the R^2 distributions, the dependence of the parameter values of the sub-models to each other is ignored. We would have many possibly true parameter values of a predictor in the same LMG comparison. It would then also be possible that the R^2 decreases when adding predictors. Another drawback is that it would be much more time-consuming to fit a separate Bayesian model for each sub-model. I therefore do not follow this path here. Using the conditional variance formula on the full-model allows to calculate LMG values in the Bayesian framework in a reasonable time exposure. Depending on the number of predictors and the number of posterior samples, the calculations still take some time in the Bayesian framework.

2.4 Bayesian regression

The following section provides a brief introduction to Bayesian regression. It further shows that assuming stochastic predictors (predictors treated as random variables) or non-stochastic predictors (predictors treated as fixed quantities) results in the same posteriors for the regression parameters under the assumption of weak exogeneity and conditional independence. It is summarized from the book *Bayesian Analysis for the Social Sciences* (Jackman, 2009). This is an important finding because it is often more appropriate to assume stochastic continuous predictors.

The assumption of weak exogeneity is introduced first. The joint probability density of $p(y_i, \boldsymbol{x}_i)$ can be factored as

$$p(y_i, \boldsymbol{x}_i \mid \boldsymbol{\theta}) = p(y_i \mid \boldsymbol{x}_i, \boldsymbol{\theta})\, p(\boldsymbol{x}_i \mid \boldsymbol{\theta}),$$

where both $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ and predictor matrix $\mathbf{X}$ of size $n \times k$ are considered random variables depending on parameter vector $\boldsymbol{\theta}$ with observations $i = 1, \ldots, n$.

The parameter vector $\boldsymbol{\theta}$ can be decomposed into two components $\boldsymbol{\theta}_{y|x}$ and $\boldsymbol{\theta}_x$. The assumption of weak exogeneity consists of the two restrictions:

$$p(y_i \mid \boldsymbol{x}_i\, \boldsymbol{\theta}) = p(y_i \mid \boldsymbol{x}_i, \boldsymbol{\theta}_{y|x})$$

and

$$p(\boldsymbol{x}_i \mid \boldsymbol{\theta}) = p(\boldsymbol{x}_i \mid \boldsymbol{\theta}_{\boldsymbol{x}}).$$

The weak exogeneity assumption implicates that the whole information about $\boldsymbol{y}_i$ is contained in $\boldsymbol{x}_i$ and $\boldsymbol{\theta}_{y|x}$. Knowledge of the parameters $\boldsymbol{\theta}_x$ provides no additional information about $\boldsymbol{y}_i$. Whether or not considering $\boldsymbol{x}_i$ as a random variable is of no consequence for learning about $\boldsymbol{\theta}_{y|x}$. When these requirements are fulfilled, $\boldsymbol{x}_i$ is said to be weakly exogenous for $\boldsymbol{\theta}_{y|x}$.

Under the assumption of conditional independence, the joint density of the data can further be written as

$$p(\boldsymbol{y}, \mathbf{X} \mid \boldsymbol{\theta}) = p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\theta}_{y|x})\, p(\mathbf{X} \mid \boldsymbol{\theta}_x),$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_{y|x}, \boldsymbol{\theta}_x)^\top$.

The interest of regression is mostly on the posterior parameters $\boldsymbol{\theta}_{y|x}$. These posterior densities are proportional to the likelihood of the data multiplied by the prior density. The joint density $p(\boldsymbol{y}, \mathbf{X} \mid \boldsymbol{\theta})$ is used to learn about the posterior parameters, via Bayes rule

$$p(\boldsymbol{\theta} \mid \boldsymbol{y}, \mathbf{X}) \propto p(\boldsymbol{y}, \mathbf{X} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta}).$$

The dependence of $\boldsymbol{y}$ on $\mathbf{X}$ is captured in the parameters $\boldsymbol{\theta}_{y|x} = (\boldsymbol{\beta}, \sigma^2)$. Under the assumption of independent prior densities about $\boldsymbol{\theta}_{y|x}$ and $\boldsymbol{\theta}_x$, the posterior distribution of the parameters can be written as

$$\begin{aligned} p(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}_x \mid \boldsymbol{y}, \mathbf{X}) &= \frac{p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2)\, p(\boldsymbol{\beta}, \sigma^2)}{p(\boldsymbol{y} \mid \mathbf{X})} \times \frac{p(\mathbf{X} \mid \boldsymbol{\theta}_x)\, p(\boldsymbol{\theta}_x)}{p(\mathbf{X})} \\ &= p(\boldsymbol{\beta}, \sigma^2 \mid \boldsymbol{y},\, \mathbf{X})\, p(\boldsymbol{\theta}_x \mid \mathbf{X}). \end{aligned} \tag{2.16}$$

The factorization in equation 2.16 shows that, under the above mentioned assumptions, the posterior inference about the parameters $\boldsymbol{\theta}_{y|x} = (\boldsymbol{\beta}, \sigma^2)$ is independent from the inference about $\boldsymbol{\theta}_x$ given data $\mathbf{X}$. This also means that the assumption about $\mathbf{X}$ being non-stochastic or stochastic results in the same posterior density of $\boldsymbol{\theta}_{y|x}$. In the case of non-stochastic regressors, $p(\mathbf{X})$ and $\boldsymbol{\theta}_x$ drop out of the calculations. For stochastic predictors, it means that, given $\mathbf{X}$, nothing more can be gained about $\boldsymbol{\theta}_{y|x} = (\boldsymbol{\beta}, \sigma^2)$ from knowing $\boldsymbol{\theta}_x$.

2.5 Stochastic or non-stochastic predictors

The focus of regression is on $\boldsymbol{\theta}_{y|x} = (\boldsymbol{\beta}, \sigma^2)$, for which it does not matter whether we assume non-stochastic or stochastic predictors under the assumptions mentioned in section 2.4. However, assuming stochastic or non-stochastic predictors influences the uncertainty of the LMG values because the variance of the predictors is also incorporated in the LMG formula. The LMG

formula may be especially interesting for continuous predictors, which often are of stochastic nature. Grömping (2006) recommends to use in most cases bootstraping for stochastic predictors when calculating bootstrap confidence intervals.

For non-stochastic predictors, the covariance of the predictors $\mathbf{X}$ is given and does not need to be estimated. The population variance, which divides the sum of squares by n and not by $n-1$, should therefore be used in definition (2.10) for non-stochastic predictors. In the frequentist setting, it does not matter by which denominator the sum of squares is divided. As long as the residual sum of squares is divided by the same denominator value as the total variance , the denominator values cancel out each other and the R^2 in definition (2.6) does not change. To directly incorporate the sample variance estimate of the σ^2 parameter in definition (2.6), both sum of squares can be divided by $n-1$. In the Bayesian framework, the σ parameter is sampled from its posterior distribution. Therefore, it makes a small difference, whether the explained sum of squares is divided by n or by $n-1$ in the Bayesian framework.

For stochastic predictors, the covariance of the predictors $\mathbf{X}$ needs to be additionally estimated. The sample covariance estimator (sum of squares dividing by $n-1$) provides an unbiased estimate of the covariance structure. However, it is just a point estimate of the covariance structure. With stochastic predictors, there is an additional uncertainty in the R^2 formula (2.10) that can have a large influence on the R^2 and the LMG values. Therefore, the information about $\boldsymbol{\theta}_x$ is also relevant for stochastic regressors. As seen in equation (2.16), inference about $\boldsymbol{\theta}_x$ is independent from inference about $\boldsymbol{\theta}_{y|x}$. If there are stochastic predictors and we only use the sample estimate of the covariance matrix, we do not incorporate the uncertainty of the estimate. Because the explained variance is calculated by $\boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}$, inference about $\boldsymbol{\theta}_x$ seems to be equally important as inference about $\boldsymbol{\theta}_{y|x}$ for stochastic predictors. Even when the exact regression parameters were known, there would be a lot of uncertainty in the LMG values caused by the uncertainty about the covariance matrix. If the distribution of the $p(\mathbf{X})$ is known, the $\boldsymbol{\theta}_x$ could be estimated. However, the computation time is then much higher, because the whole LMG calculation needs to be done for each posterior covariance sample of the predictors. Depending on the number of predictors this would be very time-consuming. In most cases, the problem is that the distribution of the $\mathbf{X}$ is unknown. As a practical solution, nonparametric bootstrapping of the covariance matrix could be used to include the uncertainty of the stochastic predictors in the LMG calculations. Again, it would be necessary to do the LMG calculations for each bootstrap sample of the covariance matrix. There exist also different covariance estimators. The shrinkage method may be an interesting estimator with some nice properties (Schäfer and Strimmer, 2005).

Chapter 3

Examples

The Bayesian LMG implementation is presented with two examples in the following chapter. The implementation code to get the R^2 values of the sub-models from the posterior distributions of the full-model can be found in Appendix A.1. Simulated data were used for the first example. Empirical data were used for the second example.

3.1 Simulated data

A simple model is assumed for the first example:

$$
\begin{aligned}
&Y_i \mid \boldsymbol{x}_i \sim \mathcal{N}(\beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \beta_4 x_{4,i},\, \sigma^2), \qquad i = 1, \ldots, 50,\\
&(\beta_1, \beta_2, \beta_3, \beta_4) = (1, 1, 1, 1), \quad \sigma^2 = 1,\\
&\mathbf{X}_m = (x_{m,1}, \ldots, x_{m,50}) \sim \mathcal{N}(0, 1), \quad m = 1, 2, 3, 4,
\end{aligned}
$$

where $i = 1, \ldots, 50$ indexes the observations and $m = 1, 2, 3, 4$ represents the four predictor variables. The data generating R-code can be found in Appendix A.2.

The model is fitted using the `rstanarm` package (Stan Development Team, 2016) with the default priors for the slope and the σ^2 parameters. These default priors are called 'weakly informative priors' because they take into account the order of magnitude of the variables by using the variance of the observed data. Information about these priors can be found in Stan Development Team (2017). The automatic scale adjustments of the default priors resulted in the following priors for the regression parameters: $\beta_0 \sim \mathcal{N}(0, 21.849)$, $\beta_1 \sim \mathcal{N}(0, 6.530)$, $\beta_2 \sim \mathcal{N}(0, 5.691)$, $\beta_3 \sim \mathcal{N}(0, 5.776)$, $\beta_4 \sim \mathcal{N}(0, 5.620)$, and σ, the error standard deviation, had an $\mathcal{E}xp(2.185)$ distribution. A burn-in period of 20 000, a sample size of 20 000, and a thinning of 20 were chosen, resulting in a posterior sample size of 1 000. The exact command can be found in R-code A.3. The posterior distributions of the parameters are shown in Figure 3.1.

For each joint posterior sample of the parameters, the R^2 value was calculated. The R^2 of the sub-models was then calculated by the conditional variance formula (2.15) for each posterior sample. The resulting R^2 values of the posterior samples are shown in Figure 3.2. The thinning is reasonable in this case to reduce the computational burden and to still obtain an appropriate posterior of the R^2 values (Link and Eaton, 2012).

The `hier.part` package was used to calculate the LMG value for each posterior sample. The independent component (I) represents the LMG value. The joint contribution (J) represents the difference from the independent component to the explained variance of the model containing only the predictor itself (T). At first, non-stochastic regressors were assumed. The resulting LMG values and joint contributions with a 95% credible interval are shown in Table 3.1.

An option to display the resulting LMG distribution is shown in Figure 3.3. Using the default weakly informative priors, the LMG distributions obtained from the Bayesian framework were very similar to the bootstrap confidence intervals, assuming non-stochastic predictors of the LMG estimates, obtained from the `relaimpo` package, as shown in Table 3.2.

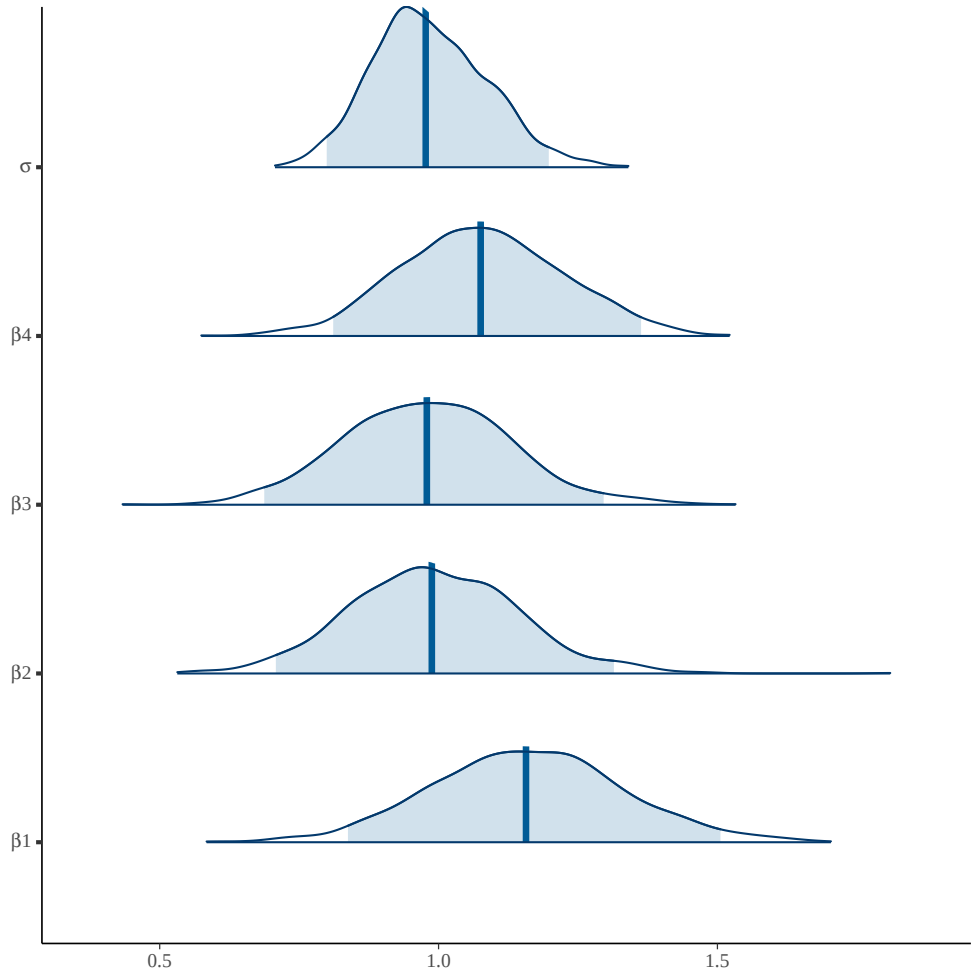

Figure 3.1: Posterior distributions of the slope and sigma parameters of the simulated data example. $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)$ represents the regression coefficients of predictors $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \mathbf{X}_4$ and σ represents the standard deviation of the conditional dependent variable distribution. The shaded area represents the 95% credible interval. The thick vertical line shows the median.

Table 3.1: Variance decomposition of the simulated data set assuming non-stochastic predictors with 95% credible interval. I = LMG value, J = joint contribution, Total = total explained variance in one-predictor-only model.

Variable	I	J	Total
$\mathbf{X}_1$	0.188 (0.097, 0.292)	-0.004 (-0.011, 0.004)	0.183 (0.095, 0.285)
$\mathbf{X}_2$	0.186 (0.101, 0.292)	0.003 (-0.002, 0.008)	0.189 (0.104, 0.298)
$\mathbf{X}_3$	0.172 (0.087, 0.270)	0.000 (-0.008, 0.009)	0.172 (0.086, 0.268)
$\mathbf{X}_4$	0.248 (0.149, 0.352)	0.028 (0.019, 0.036)	0.276 (0.173, 0.380)

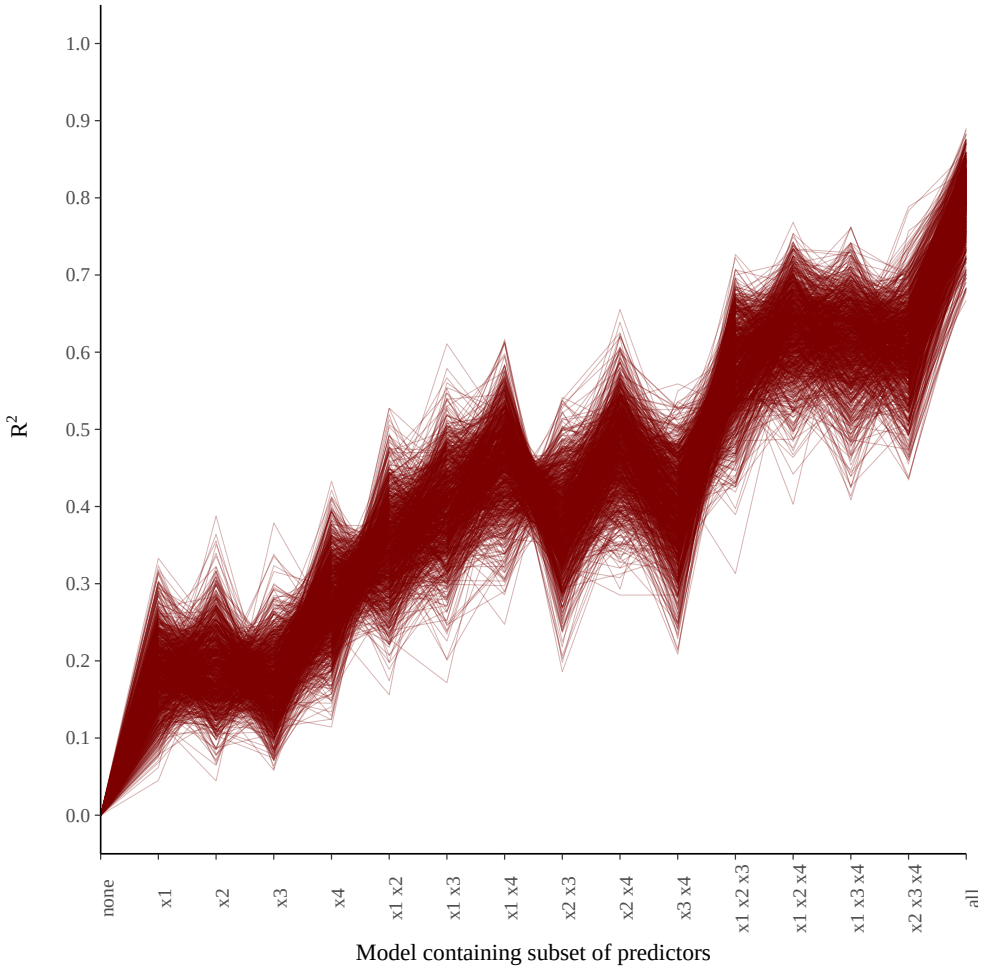

Figure 3.2: R^2 values for each posterior sample of the simulated data example containing predictors $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \mathbf{X}_4$. A connected line represents the R^2 for the full-model and for all the sub-models of these predictors for one of 1 000 posterior joint parameter samples.

Table 3.2: Comparison of Bayesian framework to frequentist framework (`relaimpo`) assuming non-stochastic predictors for the simulated data set. CI = credible interval for Bayesian approach, confidence interval for classical approach.

	LMG value (95%-CI)	
Variable	**Relaimpo**	**Bayesian framework**
$\mathbf{X}_1$	0.191 (0.111, 0.291)	0.188 (0.097, 0.292)
$\mathbf{X}_2$	0.195 (0.111, 0.294)	0.186 (0.101, 0.292)
$\mathbf{X}_3$	0.178 (0.105, 0.275)	0.172 (0.087, 0.270)
$\mathbf{X}_4$	0.257 (0.166, 0.358)	0.248 (0.149, 0.352)

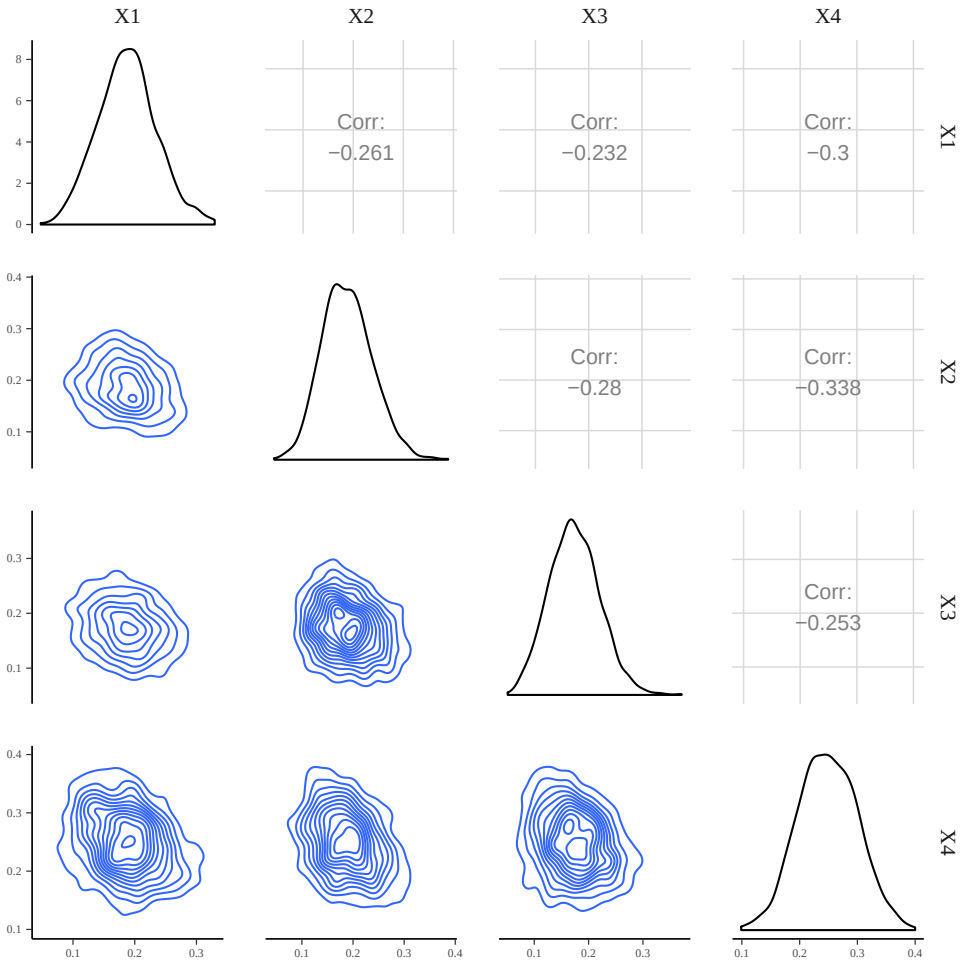

Figure 3.3: LMG distributions of the four predictors $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$, and $\mathbf{X}_4$ of the simulated data set assuming non-stochastic predictors. Corr. = correlation between LMG distributions.

In this example with simulated data, we know that the predictor values were sampled from a normal distribution. It would therefore be more realistic to assume stochastic predictors. As described in the theory section 2.4 and 2.5, the posterior distributions of the regression parameters $\boldsymbol{\beta}$ are valid for non-stochastic and stochastic predictors under the assumption of weak exogeinity and conditional independence. However, the uncertainty about the LMG values needs to include the uncertainty about the covariance matrix. If we know that the distribution of the predictors $\mathbf{X}$ is a multivariate normal distribution with covariance matrix $\boldsymbol{\Sigma}$, we can obtain the posterior distribution of its covariance matrix $\boldsymbol{\Sigma}$. This information can then be incorporated in the R^2 calculations as described in the theory section. The package `JAGS` (Plummer, 2017) was used for inference about the covariance matrix in a Bayesian way. It was assumed that the predictor values $\mathbf{X}_i$ from each observation i were distributed as $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. For each element of $\boldsymbol{\mu}$ a $\mathcal{N}(0, 4)$ prior distribution was chosen. For the inverse covariance matrix $\boldsymbol{\Sigma}^{-1}$, a Wishart distribution prior, $\mathcal{W}(\mathbf{I}, n)$, where $\mathbf{I}$ is the identity matrix of size 4×4 and $n = 50$, was chosen. The R-code of the covariance inference can be found in Appendix A.4. As an alternative, non-parametric bootstrap was used for inference about the covariance matrix. The R-code of the bootstrap implementation can be found in Appendix A.5.

In contrast to non-stochastic predictors, the uncertainty about the covariance matrix is reflected in the larger credible intervals for stochastic predictors. Table 3.3 shows the LMG values of the different approaches. Either using the bootstrap samples of the covariance matrix or using samples from the posterior covariance matrix produced very similar LMG distributions. Bootstrap seems to be a valuable option for stochastic predictors when the distribution of the predictors is unknown. Even when the distribution is known, the difference seems to be tiny. A benefit of going the full Bayesian way is that prior knowledge about the covariance matrix can also be included. Using the default priors further produced very similar LMG distribution as using the non-parametric bootstrap option of the `relaimpo` package.

Table 3.3: LMG values of different approaches assuming stochastic predictors for the simulated data set with 95% CI (credible intervals for Bayesian approaches, confidence interval for frequentist approach (`relaimpo`)).

	Frequentist framework	**Bayesian framework**	
Variable		non-parametric bootstrap	covariance inference
$\mathbf{X}_1$	0.191 (0.061, 0.353)	0.186 (0.072, 0.331)	0.187 (0.075, 0.331)
$\mathbf{X}_2$	0.195 (0.075, 0.338)	0.184 (0.073, 0.327)	0.187 (0.075, 0.335)
$\mathbf{X}_3$	0.178 (0.079, 0.298)	0.169 (0.060, 0.310)	0.171 (0.067, 0.309)
$\mathbf{X}_4$	0.257 (0.135, 0.424)	0.243 (0.121, 0.377)	0.244 (0.116, 0.384)

3.2 Empirical data

In the following section, the Bayesian LMG implementation is applied on an empirical dataset containing test scores of pupils (N = 301) from a study by Holzinger and Swineford (1939) available in the R package `MBESS` (Kelley, 2017). This dataset was used in Nimon *et al.* (2008) to present commonality analysis, which is another variance decomposition technique. Scores from a paragraph comprehension test (paragrap) were predicted by four verbal tests: general-information (general), sentence-comprehension (sentence) , word-classification (wordc), and word-meaning (wordm) (Table 3.4).

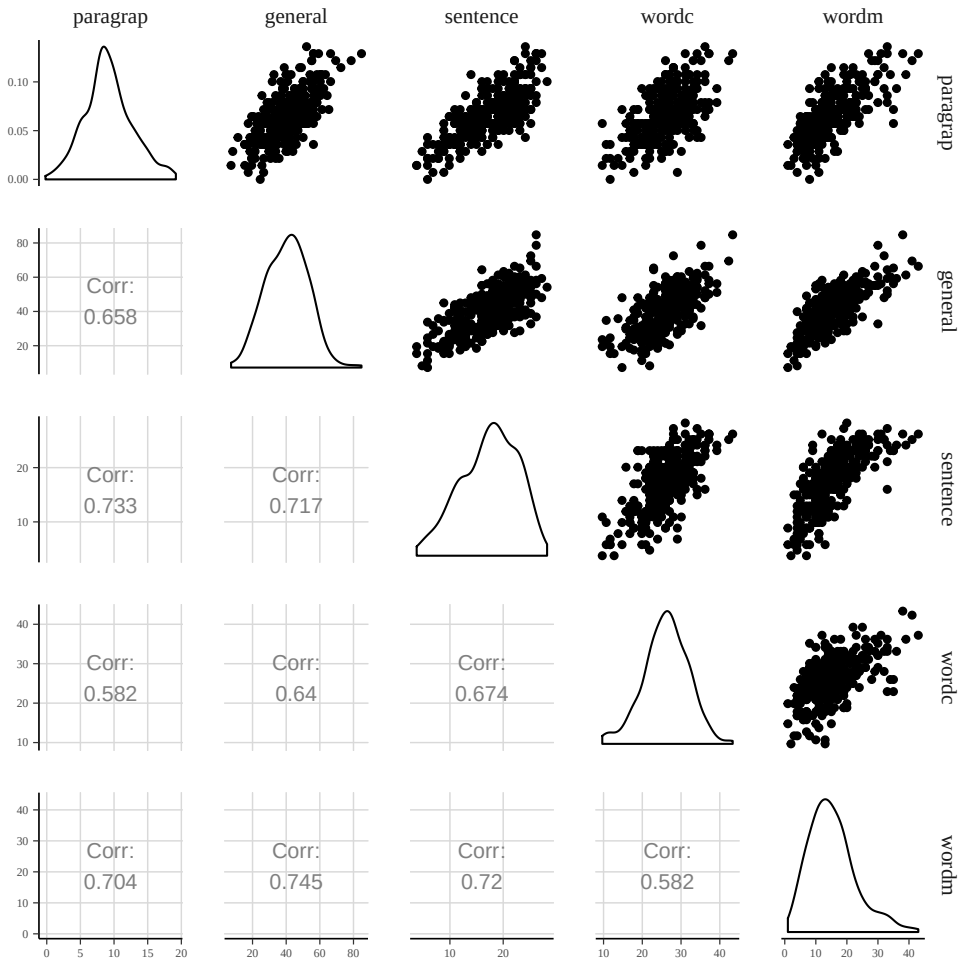

Figure 3.4: Empirical data set. Test scores from Holzinger and Swineford (1939) Study. N=301. The variable description can be found in Table 3.4.

The aim of the regression analysis was to determine the association between verbal ability and paragraph comprehension. An overview of the data is shown in Figure 3.4. The regression results from a simple linear regression model including all four predictors are shown in Table 3.5. A novice researcher may wrongly conclude, that there is little association between the "non-significant" predictors (general information and word-classification) and paragraph comprehension. Given the other predictors are already included in the model, the predictors seem not to provide much information about the expected paragraph comprehension ability. However, it should not be concluded from this regression table that there is no association between any of these "non-significant" predictors and the dependent variable. As shown in Figure 3.4, the correlations between the predictors are rather high. The LMG metric may therefore provide new information about the importance of each predictor.

Table 3.4: Variable description of the empirical data set.

Variable	Description
paragrap	scores on paragraph comprehension test
general	scores on general information test
sentence	scores on sentence completion test
wordc	scores on word classification test
wordm	scores on word meaning test

The Bayesian regression model was fitted in `rstanarm`. The default priors were used for the slope coefficients and the σ^2 parameter. The automatic scale adjustments of the default priors resulted in the following priors for the regression parameters: $\beta_0 \sim \mathcal{N}(0, 34.923)$, $\beta_1 \sim \mathcal{N}(0, 0.700)$, $\beta_2 \sim \mathcal{N}(0, 1.691)$, $\beta_3 \sim \mathcal{N}(0, 1.538)$, $\beta_4 \sim \mathcal{N}(0, 1.138)$, and σ, the error standard deviation, has an $\mathcal{E}xp(3.492)$ prior distribution. A burn-in period of 20 000, a sample size of 20 000, and a thinning of 20 resulted in a posterior sample size of 1 000. The exact commands can be found in R-code A.6. The posterior distribution of the slope regression parameters are shown in Figure 3.5. The resulting R^2 of these posterior samples are shown in Figure 3.6. The LMG values were calculated by using `hier.part`. The independent component (I), the joint contribution (J), and the total explained variance in a one-predictor-only model (T) are shown in Table 3.6. The LMG distributions are displayed in Figure 3.7. Sentence-comprehension and word-meaning seem to be the most important predictors by applying the LMG metric. However, none of the predictors seem to be unimportant. The joint contributions of each predictor were quite large.

For comparison purposes, the LMG metric was additionally calculated with the `relaimpo` package using parametric bootstrapping. The confidence intervals of `relaimpo` were almost identical to the credible intervals of the Bayesian framework (Table 3.7). Assuming stochastic or non-stochastic predictors resulted also in almost identical uncertainty estimates with such a large sample size ($N = 301$) because the covariance matrix was estimated with high precision for the stochastic predictors (Table 3.8). Because the computation time is much larger for stochastic predictors, it can be much more efficient for large sample sizes to use the sample point estimate of the covariance matrix (in contrast to incorporating a distribution of possible covariance values) in the LMG calculations even when stochastic predictors are assumed.

Table 3.5: Regression output of paragraph comprehension on verbal tests. The variable description of the predictors can be found in Table 3.4.

	Coefficient	95%-confidence interval	p-value
Intercept	0.071	from -1.17 to 1.31	0.91
general	0.03	from -0.00 to 0.06	0.084
sentence	0.26	from 0.18 to 0.34	< 0.0001
wordc	0.047	from -0.01 to 0.11	0.14
wordm	0.14	from 0.08 to 0.19	< 0.0001

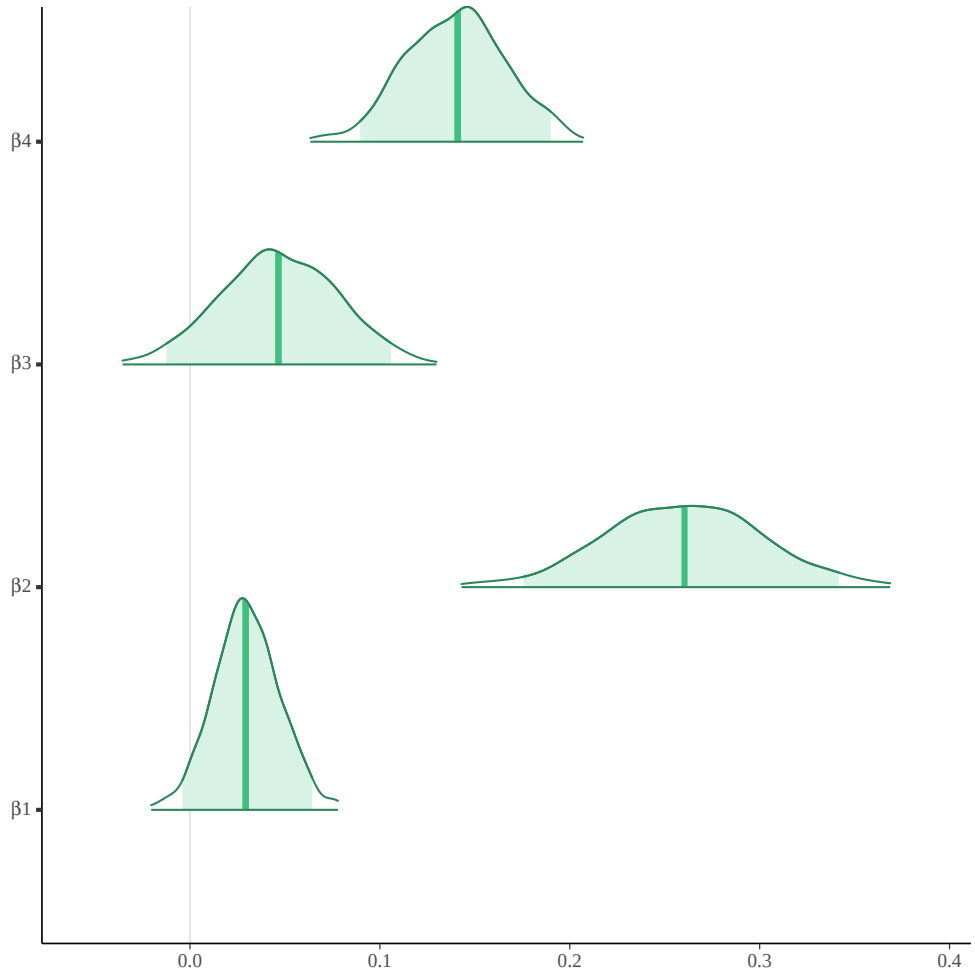

Figure 3.5: Posterior distributions of the slope parameters of the different verbal ability predictors. $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)$ represents the regression coefficients of the four predictors *general*, *sentence*, *wordc*, *wordm*. The variable description of the predictors can be found in Table 3.4. The shaded area represents the 95% credible interval. The thick vertical line shows the median.

Table 3.6: Variance decomposition of the empirical data set assuming non-stochastic predictors with 95% credible intervals. I = LMG values, J = joint contribution, Total = total explained variance in one-predictor-only model.

Variable	I	J	Total
general	0.130 (0.104, 0.160)	0.298 (0.259, 0.332)	0.429 (0.364, 0.488)
sentence	0.203 (0.162, 0.245)	0.327 (0.292, 0.358)	0.530 (0.463, 0.588)
wordc	0.095 (0.074, 0.122)	0.238 (0.202, 0.274)	0.334 (0.276, 0.394)
wordm	0.178 (0.142, 0.216)	0.315 (0.281, 0.345)	0.492 (0.430, 0.554)

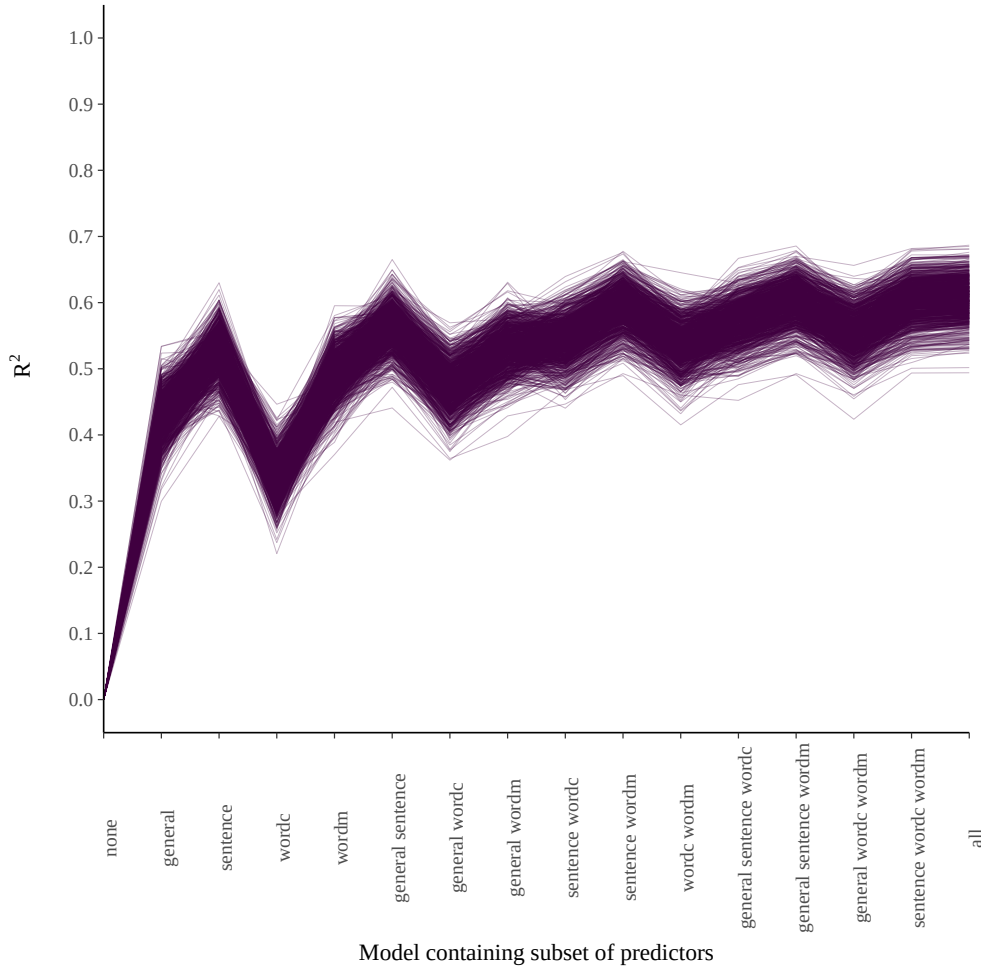

Figure 3.6: R^2 values for each posterior sample of the empirical data example. The variable description of the four predictors *general*, *sentence*, *wordc*, *wordm* can be found in Table 3.4. A connected line represents the R^2 for the full-model and for all the sub-models of these predictors for one of 1 000 posterior joint parameter samples.

Table 3.7: Comparison of Bayesian framework to frequentist framework (`relaimpo`) assuming non-stochastic predictors for the empirical data set. CI = credible interval for Bayesian framework and confidence interval for frequentist approach.

	LMG value (95%-CI)	
Variable	**Frequentist framework**	**Bayesian framework**
general	0.131 (0.104, 0.162)	0.130 (0.104, 0.160)
sentence	0.206 (0.168, 0.247)	0.203 (0.162, 0.245)
wordc	0.097 (0.074, 0.127)	0.095 (0.074, 0.122)
wordm	0.178 (0.142, 0.219)	0.178 (0.142, 0.216)

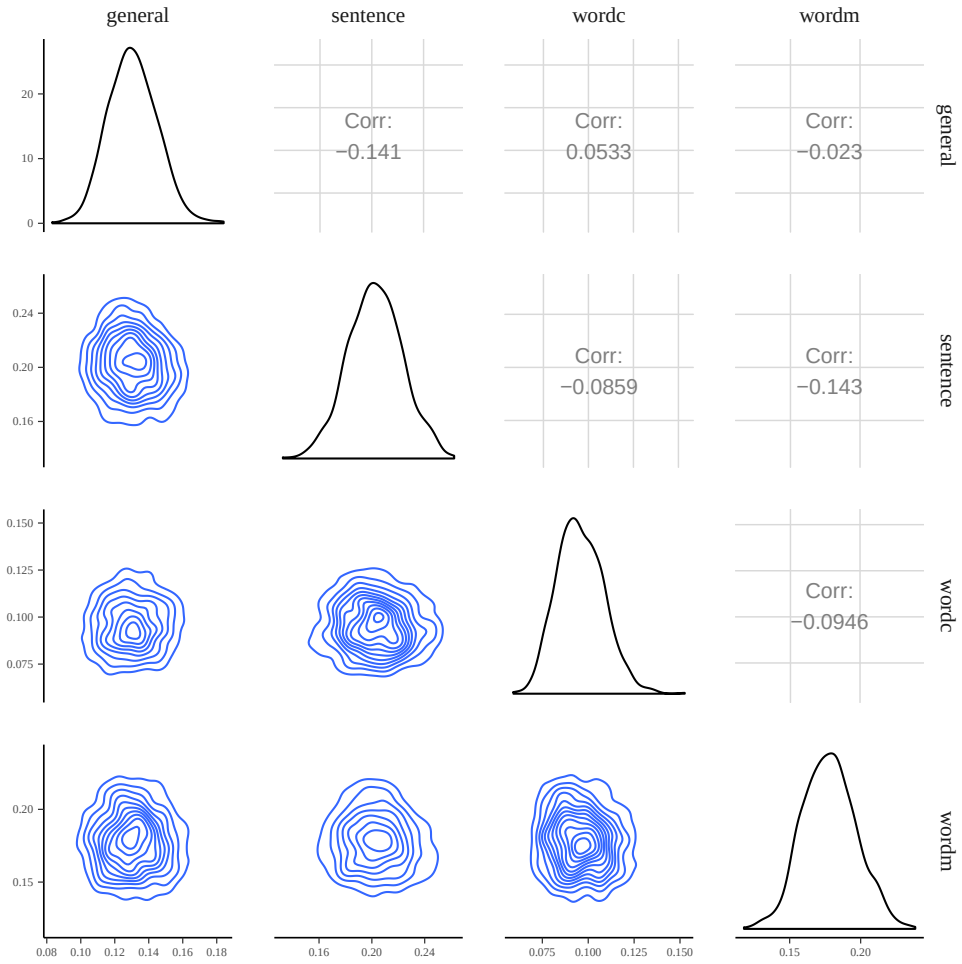

Figure 3.7: LMG distributions of the different verbal ability predictors. The variable description of the predictors can be found in Table 3.4. Corr. = correlation between LMG distributions.

Table 3.8: Comparison of Bayesian framework to frequentist framework (`relaimpo`) assuming stochastic predictors for the empirical data set. CI = credible interval for Bayesian framework and confidence interval for frequentist approach.

	LMG value (95%-CI)	
Variable	**Frequentist framework**	**Bayesian framework**
general	0.131 (0.106, 0.160)	0.130 (0.099, 0.163)
sentence	0.206 (0.170, 0.250)	0.202 (0.161, 0.245)
wordc	0.097 (0.074, 0.127)	0.095 (0.067, 0.126)
wordm	0.178 (0.141, 0.221)	0.177 (0.140, 0.218)

Chapter 4

Extension to longitudinal data

Some extensions of the LMG formula beyond the simple linear regression model are shown in the following chapter. The focus is on repeated measures models. These models extend the simple linear regression by allowing intra-subject correlation between repeated measures.

The dependence of within-subject measurements can be modeled by including random effects (mixed model) or by assuming correlated errors within a subject (marginal model). A mixed model can be extended by including a random slope per subject, allowing for more general longitudinal shapes. Different covariance matrices of the error terms allow for more general longitudinal shapes in the marginal approach. An unstructured covariance matrix, where no restriction is imposed, allows for the most freedom in the error term. However, depending on the number of repeated measurements and the sample size, the covariance matrix can get too large to make feasible inference (Fitzmaurice *et al.*, 2011).

The extension of the LMG formula in the Bayesian framework applied to longitudinal models in this master thesis is restricted to models where the conditional variance formula can be applied to obtain the explained variance of the sub-model from the regression parameters of the full-model. Therefore, the focus of this thesis is on the relative importance of the fixed effects, but not on the relative importance of the random effects. The conditional variance formula can be used in the marginal models, because only the fixed effects are modeled anyway. In the mixed model framework, the conditional variance formula is applicable to random intercept models. For random-slope models, there are at least some difficulties involved – if it is possible at all – to obtain the explained variance of the sub-models.

In this chapter, the Bayesian LMG implementation is shown on a random intercept model and on a repeated measures model with an unstructured covariance matrix.

4.1 Random intercept model

The first example concerns a simple random intercept model with time-varying predictors. There exist different R^2 metrics for linear mixed models (Nakagawa and Schielzeth, 2013). The variance of a random intercept model with regression parameter $\boldsymbol{\beta}$ can be written as

$$\mathrm{Var}(y) = \sigma_f^2 + \sigma_\alpha^2 + \sigma_\epsilon^2, \tag{4.1}$$

where σ_α^2 is the variance of the random intercept, σ_ϵ^2 represents the error variance and $\sigma_f^2 = \mathrm{Var}(\mathbf{X}\boldsymbol{\beta}) = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}$ with regression parameters $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ without the intercept of size $p \times 1$ and predictor matrix $\mathbf{X}$ of size $n \times p$ with corresponding covariance matrix $\boldsymbol{\Sigma}_{\mathbf{XX}}$ of size $p \times p$. An R^2 that is guaranteed to be positive is defined in Nakagawa and Schielzeth (2013) as

$$R^2_{\mathrm{LMM}} = \frac{\sigma_f^2}{\sigma_f^2 + \sigma_\alpha^2 + \sigma_\epsilon^2}. \tag{4.2}$$

Theoretically, it is possible that the R^2_{LMM} decreases when adding predictors (Nakagawa and Schielzeth, 2013). However, by adding predictors, σ^2_f should always increase but σ^2_ϵ or σ^2_α may also increase and the total R^2 may then be lower. The R^2 cannot decrease by using the conditional variance formula (2.15) on the full-model to calculate the R^2 of the sub-models, because the total variance is fixed. In the maximum likelihood framework, use of the conditional variance formula on the maximum likelihood parameter estimates of the full-model should lead to equal results that would be obtained by first fitting a new model by maximum likelihood for each subset of predictors and afterwards comparing the explained variance of the fixed effects of these sub-models to the total variance of the full-model. In the Bayesian framework, the conditional variance formula is necessary to account for the mutual interdependence of the sub-models. The total variance of the full-model can be calculated as $\text{Var}(y) = \text{Var}(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}b) + \sigma^2$ or by using samples of σ^2_α as in definition (4.1). The error term could again be sampled or calculated as in definition (2.9). In the following examples, definition (4.1) was used, σ^2_α, σ^2_ϵ, and $\boldsymbol{\beta}$ were sampled from their posterior distribution.

In repeated measures studies, the focus is often on within-subject changes. The between-subject variance, estimated with the random intercept term, is of minor importance. The important question often is how much variance do the fixed predictors explain, compared to the variance of the within-subject error, which is

$$R^2_{\text{repeated}} = \frac{\sigma^2_f}{\sigma^2_f + \sigma^2_\epsilon}, \tag{4.3}$$

The square root of this term is known under the name *correlation within subjects* in Bland and Altman (1995). Often, there are between-subject and within-subject predictors in a model. If the interest lies in the within-subject effects, a model including only the between-subject predictors can be used as the null-model.

The following example shows a simple random intercept model with time-varying predictors. The main question was which within-subject predictors were the most important ones. The between-subject variance was of minor importance. The data were simulated from the following regression setting with $m = 4$ timepoints and $n = 20$ number of subjects (see Appendix A.7 for R-code),

$$Y_{i,j} \sim \mathcal{N}(\beta_0 + \beta_1 x_{1_{i,j}} + \beta_2 x_{2_{i,j}} + \beta_3 x_{3_{i,j}} + \beta_4 x_{4_{i,j}} + \alpha_i,\, \sigma^2), \qquad \begin{aligned} i &= 1, \ldots, n, \\ j &= 1, \ldots, m, \end{aligned}$$

where $(\beta_1, \beta_2, \beta_3, \beta_4) = (1, 1, 2, 2)$, $\sigma^2 = 1$, $\alpha_i \sim \mathcal{N}(0, \sigma^2_\alpha)$ with $\sigma^2_\alpha = 16$, and $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.3 & 0.4 & 0.4 \\ 0.3 & 1 & 0.4 & 0.4 \\ 0.4 & 0.4 & 1 & 0.9 \\ 0.4 & 0.4 & 0.9 & 1 \end{pmatrix}.$$

The individual trajectories are shown in Figure 4.1. The random intercept effect was of minor interest. The Bayesian R^2 of the models was calculated according to the formula of repeated measure correlation (4.3) using the conditional variance formula (2.15). The R-code of the model can be found in Appendix A.8. Most of the within-subject variance was explained by the predictors (Table 4.1). The credible intervals were very small because non-stochastic predictors were assumed and because the within subject error σ^2_ϵ was very small compared to the explained variance of the predictors. For information about the between-subject variance term, we can look at the posterior distribution of the random intercept variance term σ^2_α.

Next, the random intercept was directly included in the total variance calculation of the R^2 values. The R-code can be found in the Appendix A.9. There was a large between-subject variance ($\sigma^2_\alpha = 16$) in this simulated data set. Therefore, the LMG values including the between

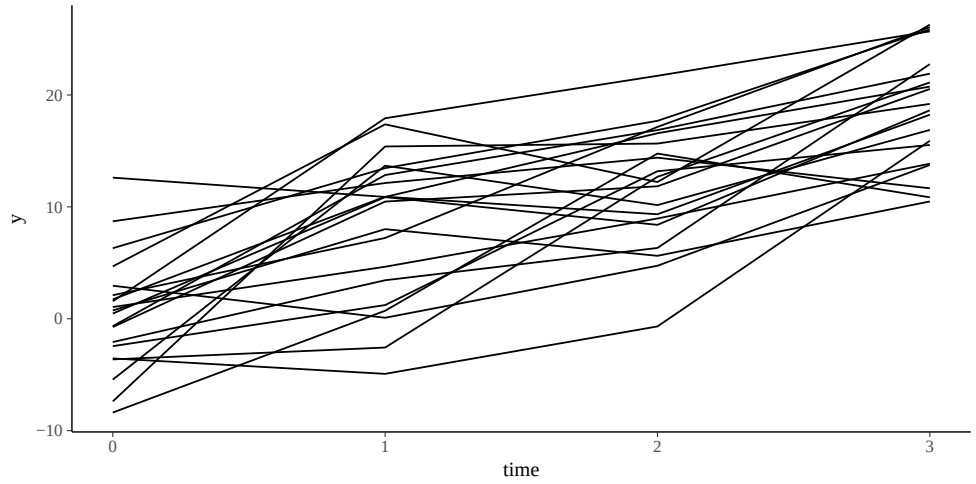

Figure 4.1: Individual trajectories of simulated random intercept model.

subject variance were much lower (Table 4.2). The credible intervals were also much larger, because the uncertainty about the between-subject variance was included.

In my opinion, we can obtain more useful information from separating the between-subject variance from the within-subject variance in this simple case. This makes it possible to quantify the uncertainty of the between-subject variance and the within-subject variance. Definition (4.2) and definition (4.3) do not exclude each other. Both can provide useful information. Note that we assumed non-stochastic predictors. Otherwise, the credible intervals would again be larger (results not shown). In general, it seems more reasonable to assume stochastic time-varying continuous predictors because we only have a sample of the population. The exact predictor values vary from person to person. In the case of stochastic predictors, the variance could be estimated by non-parametric bootstrap, resampling whole subjects (all repeated measurements of a subject).

Table 4.1: Variance decomposition with focus on whithin-subject explained variance of simulated random intercept data with 95% credible intervals. I = LMG values, J = joint contribution, Total = total explained variance in one-predictor model.

Variable	I	J	Total
	0.204 (0.203, 0.206)	0.515 (0.514, 0.517)	0.720 (0.716, 0.723)
	0.190 (0.189, 0.192)	0.491 (0.489, 0.493)	0.682 (0.678, 0.685)
	0.299 (0.297, 0.300)	0.628 (0.627, 0.628)	0.927 (0.925, 0.928)
	0.306 (0.305, 0.308)	0.638 (0.638, 0.638)	0.944 (0.943, 0.946)

Table 4.2: Variance decomposition with focus on total explained variance of simulated random intercept data with 95% credible intervals. I = LMG values, J = joint contribution, Total = total explained variance in one-predictor model.

Variable	I	J	Total
	0.149 (0.112, 0.175)	0.377 (0.282, 0.443)	0.527 (0.394, 0.618)
	0.139 (0.104, 0.163)	0.360 (0.269, 0.422)	0.499 (0.373, 0.585)
	0.219 (0.163, 0.256)	0.460 (0.343, 0.539)	0.679 (0.507, 0.794)
	0.224 (0.168, 0.263)	0.467 (0.349, 0.547)	0.692 (0.516, 0.810)

4.2 Marginal model

The next example is about a repeated measurement model with time-varying predictors and an unstructured error covariance matrix. The data were generated from the following model

$$Y_i \sim \mathcal{N}(\mathbf{X}_i\boldsymbol{\beta}, \boldsymbol{\Sigma}), \qquad i = 1, \ldots, 50, \tag{4.4}$$

where $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4) = (1, 1, 2, 2)$, $\mathbf{X}_i$ represents the predictor matrix of size 4×4 of subject i generated from $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_X)$ with

$$\boldsymbol{\Sigma}_X = \begin{pmatrix} 1 & 0.3 & 0.4 & 0.4 \\ 0.3 & 1 & 0.4 & 0.4 \\ 0.4 & 0.4 & 1 & 0.9 \\ 0.4 & 0.4 & 0.9 & 1 \end{pmatrix}, \text{ and } \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 5 & 4 & 3 & 3 \\ 4 & 7 & 3 & 3 \\ 3 & 3 & 10 & 8 \\ 3 & 3 & 8 & 10 \end{pmatrix}$$

represents an unstructured error covariance matrix, and $i = 1, \ldots, 50$ indexes the observations.

The R-code of the data generation can be found in Appendix A.10. In the variance calculation, it is necessary to take into account that there is not just one σ^2 parameter but a covariance matrix $\boldsymbol{\Sigma}$. The diagonal elements of $\boldsymbol{\Sigma}$ represent the variance of each timepoint. The trace of the covariance matrix provides the residual sum of squares per subject. In other words, the trace of $\boldsymbol{\Sigma}$ divided by the number of timepoints multiplied by the number of timepoints provides the residual sum of squares per subject. Therefore, the trace of $\boldsymbol{\Sigma}$ divided by the number of timepoints m can be taken to make the formula compatible with the $\boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}$ of (2.10), resulting in the total variance term

$$\text{Var}(y) = \sigma_f^2 + \frac{\text{tr}(\boldsymbol{\Sigma})}{m}, \tag{4.5}$$

where m represents the number of timepoints, $\boldsymbol{\Sigma}$ represents the unstructured covariance matrix of size $m \times m$, and $\sigma_f^2 = \text{Var}(\mathbf{X}\boldsymbol{\beta}) = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_{\mathbf{XX}} \boldsymbol{\beta}$ with regression parameters $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ without the intercept of size $p \times 1$ and predictor matrix $\mathbf{X}$ of size $n \times p$ with corresponding covariance matrix $\boldsymbol{\Sigma}_{\mathbf{XX}}$ of size $p \times p$.

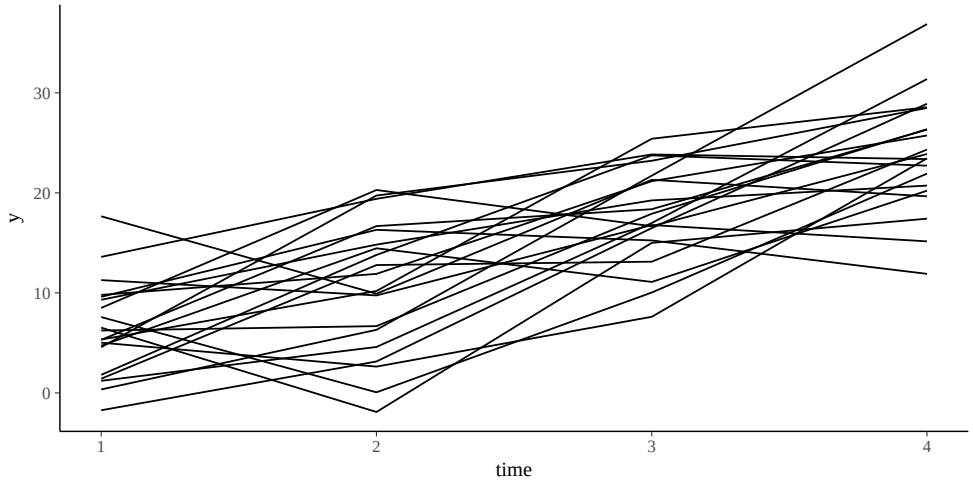

Figure 4.2: Individual trajectories of simulated marginal model data with unstructured error covariance matrix.

The individual trajectories are shown in Figure 4.2. The R-code for the model can be found in Appendix A.11. The resulting LMG values of the predictors are shown in Table 4.3.

Table 4.3: Variance decomposition for predictors of marginal model data with 95% credible intervals. I = LMG values, J = joint contribution, Total = total explained variance in one-predictor model.

Variable	**I**	**J**	**Total**
$\mathbf{X}_1$	0.077 (0.035, 0.124)	0.186 (0.084, 0.296)	0.263 (0.120, 0.420)
$\mathbf{X}_2$	0.074 (0.034, 0.117)	0.183 (0.082, 0.286)	0.257 (0.116, 0.403)
$\mathbf{X}_3$	0.115 (0.052, 0.180)	0.243 (0.110, 0.382)	0.358 (0.162, 0.563)
$\mathbf{X}_4$	0.117 (0.052, 0.184)	0.242 (0.110, 0.382)	0.358 (0.163, 0.567)

Chapter 5

Conclusion

The Bayesian framework provides the option to include prior information about parameters. Using the conditional variance formula allows to calculate the R^2 of all the sub-models from the posterior parameter distributions of the full-model. Instead of fitting $2^p - 1$ models, only the full-model needs to be fitted. The mutual interdependence of the sub-models is then automatically respected. The R^2 of the sub-models do not decrease when adding predictors. The important property of non-negativity shares is then respected in the Bayesian framework.

A disadvantage about calculating the R^2 of all the sub-models with the conditional variance formula seems to be the restriction to the linear model. Although, this may be a topic of further research. Another disadvantage of the Bayesian framework (compared to the classical LMG implementation) are higher computational costs. The calculations are still possible in a reasonable time period for non-stochastic predictors when parallel computing is used. For stochastic predictors, the computation time is much higher than in the classical framework.

Assuming non-stochastic or stochastic predictors can have a big impact in small samples on the uncertainty of the explained variance and on the LMG values. Although the posterior regression parameter distributions are the same in both cases (under the assumptions described in section 2.4), the explained variance of a model is directly dependent on the covariance matrix. Inference about the covariance of the predictors $\mathbf{X}$ is therefore an important part when stochastic predictors are assumed. However, this does not seem to be an easy problem in general. If the distribution of the predictors is known, the posterior distribution of the predictor covariance matrix can be obtained in a Bayesian way. The uncertainty about the covariance matrix can then be incorporated in the LMG calculations. However, the distribution of the predictors is often unknown in real-world data. Non-parametric bootstrap of the covariance matrix provides a practical solution in the Bayesian framework.

When the sample size is large enough, the classical and the Bayesian framework should lead to very similar values. The Bayesian framework allows including prior information, that may especially be relevant for small sample sizes. The credible interval of the Bayesian framework, in contrast to the confidence intervals, may further be easier to interpret in the mathematically correct way.

A lot of studies are concerned with within-subject changes. The extension of the LMG formula to those kinds of problems is not straightforward. It is depending on the complexity of the data. However, the extension seems to be easily possible when the focus is on the fixed effects for the simple random intercept model. When the focus is on within-subject effects, it seems reasonable to compare only the explained variance of the fixed effects of the sub-models against the variance components $\sigma_f^2 + \sigma_\epsilon^2$ obtained with the full-model. When the focus is on the total explained variance, the explained variance of the fixed effects of the sub-models can be compared against the total variance $(\sigma_f^2 + \sigma_\alpha^2 + \sigma_\epsilon^2)$ obtained with the full-model. Otherwise, there may be problems with the non-negativity property of the shares.

Appendix A

R-codes

A.1 Implementation to calculate R^2 of sub-models from posterior samples

R-Code A.1: Function to obtain R^2 from posterior samples of the full-model

```
#' An all-subset Rtwo function for the linear Bayesian regression model
#'
#' This function  calculates R^2 values for all sub-models for each posterior sample
#' with the option to include the uncertainty about the covariance matrix
#' for stochastic predictors by incorporating samples from the covariance matrix.
#' @param df  Predictor data as data frame
#' @param post.betas posterior samples of regression parameters as matrix
#' @param post.sigmas posterior samples of sigma parameters (standard deviation) as matrix
#' @param boot.M option to provide samples of predictor covariance matrix (bootstrap or Bayesian inference)
#' matrix of size m x m x boot.n, where m = number of predictors
#' @keywords LMG
#' @export
#' @examples
#' allSubsetRtows()


allSubsetRtwos <- function(df, post.betas, post.sigmas, boot.M = NULL) {

        #------- Prepare dataframe and rownames of submodels-----------------------------------------------------

        X <- cov(df) * (nrow(df) - 1) / nrow(df) # used vor variable names and as covariance of predictors for non-stochastic predictors

        lst <- list()
        pcan <- dim(df)[2] # number of predictors
        n <- (2^pcan) - 1 # number of different subsets

        for (i in 1:pcan) {
                lst[[i]] <- combn(pcan, i)   # indices of all possible subsets of predictors
        }

        var.names <- character(length = 0) # create vector to fill later with variable names
        v <- rownames(X) # names of predictors

        # create variable names of all possible subsets of predictors
        for (i in 1:length(lst))
        {
                for (j in 1:ncol(lst[[i]])) {
                        cur <- lst[[i]][, j]
                        name <- paste0(v[-cur])
                        name <- paste(name, collapse = " ")
                        var.names <- c(var.names, name)
                }
        }

        var.names <- c(rev(var.names), "all") # reverse order such that the full-model is at the bottom
        var.names[1] <- "none"
        size <- nrow(post.betas) # number of parameter samples
        df.Rtwos <- matrix(0, n + 1, 1) # prepare matrix to save R^2
        rownames(df.Rtwos) <- var.names
        # data frame preparation finished


        #-----------------Calculate R2 of all sub-models-----------------------------------------------
        # conditional variance formula:The explained variance of a sub-model can be calculated by
        # subtracting the explained variance of the not-in-the-model-included predictors that
        # is not explained by in-the-model-included predictors from the total explained variance.


        v <- 1:dim(X)[2] # need later in the loop
```

```
        post.sigmas <- as.matrix(post.sigmas) #to make sure that the sigma parameters are in matrix format
        post.betas <- as.matrix(post.betas) # to make sure that the slope parameters are in matrix format

        #-----------------For non-stochastic predictors-------------------------------------------------------
        # If non-stochastic predictor are assumed uncertainty of the covariance matrix does not need to be included
        # This part gets excecuted if no covariance samples are given.
        if (missing(boot.M) || is.na(dim(boot.M)[3])) {  #check for covariance samples
                foreach(s = 1:size, .combine = cbind) %dopar% { #for each joint posterior parameter sample
                        sample.s <- post.betas[s, ]
                        tot.var.explain <- sample.s %*% X %*% sample.s # total explained variance of the predictors
                        count <- n  # indices of the sub-models

                        # calculate R^2 values of all sub-models
                        for (i in 1:(length(lst) - 1)) # iterate over the list of sub-models
                        {
                                for (j in 1:ncol(lst[[i]])) {
                                        cur <- lst[[i]][, j] # indices of the not-in-the-model-included predictors
                                        set <- v[-cur] # indices of the in-the-model-included predictors
                                        matr <- X[cur, cur] - X[cur, set] %*% solve(X[set, set]) %*% X[set, cur] # conditional variance formula
                                        # multiply covariance matrix by parameter sample to obtain
                                        # variance not explained by in-the-model-included predictors but that can be
                                        # explained by not-in-the-model included predictors
                                        var.not.explain <- sample.s[cur] %*% matr %*% sample.s[cur]
                                        # explained variance of sub-model: total explained variance minus explained variance that can
                                        # only be explained by not in-the-model included predictors
                                        df.Rtwos[count] <- tot.var.explain - var.not.explain

                                        count <- count - 1
                                }
                        }
                        df.Rtwos[n + 1] <- tot.var.explain
                        df.Rtwos <- df.Rtwos / c(sum(c(tot.var.explain, post.sigmas[s, ]^2))) # normalize by total variance of y
                }
        }


        #------------------ For stochastic predictors-------------------------------------------------------
        #(include uncertainty about covariance infernce by incorporating samples from the covariance matrix)
        #--------------------------------------------------------------------------------------------------
        else {
                boot.n <- dim(boot.M)[3] # number of covariance samples (need to iterate over each covariance sample)
                foreach(b = 1:boot.n, .combine = cbind) %:% # iterate over each covariance sample
                        foreach(s = 1:size, .combine = cbind) %dopar% { # iterate over each joint posterior parameter sample
                                # same as code for non-stochastic predictors for each covariance sample
                                X <- boot.M[, , b]
                                sample.s <- post.betas[s, ]
                                tot.var.explain <- sample.s %*% X %*% sample.s # total explained variance of the predictors
                                count <- n # indices of the sub-model

                                # calculate R^2 values of all sub-models
                                for (i in 1:(length(lst) - 1)) #iterate over the list of sub-models
                                {
                                        for (j in 1:ncol(lst[[i]])) {
                                                cur <- lst[[i]][, j] # indices of the not-in-the-model-included predictors
                                                set <- v[-cur] # indices of the in-the-model-included predictors
                                                matr <- X[cur, cur] - X[cur, set] %*% solve(X[set, set]) %*% X[set, cur] # conditional variance
                                                # multiply covariance matrix by parameter sample to obtain
                                                # variance not explained by in the model included predictors that
                                                # can be explained by not-in-the-model included predictors
                                                var.not.explain <- sample.s[cur] %*% matr %*% sample.s[cur]
                                                # explained variance of sub-model: total explained variance - explained variance that
                                                # can only be explained by not in-the-model included predictors
                                                df.Rtwos[count] <- tot.var.explain - var.not.explain

                                                count <- count - 1
                                        }
                                }
                                df.Rtwos[n + 1] <- tot.var.explain
                                df.Rtwos <- df.Rtwos / c(sum(c(tot.var.explain, post.sigmas[s, ]^2))) # normalize by total variance of y
                        }
        }
}
```

A.2 Code used in chapter 3

A.2.1 Simulated data example

R-Code A.2: Data generation of simulated data example

```
# Data generation of simulated data example

# predictor values
x1 <- rnorm(50, 0, 1)
x2 <- rnorm(50, 0, 1)
x3 <- rnorm(50, 0, 1)
x4 <- rnorm(50, 0, 1)

# regression parameters
b1 <- 1
b2 <- 1
b3 <- 1
b4 <- 1

# dependent variable
y <- b1 * x1 + x2 * b2 + b3 * x3 + b4 * x4 + rnorm(50, 0, 1)

df <- data.frame(y = y, x1 = x1, x2 = x2, x3 = x3, x4 = x4)
```

R-Code A.3: LMG calculations for non-stochastic predictors

```
# run regression model in rstanarm with default priors
post2 <- stan_glm(y ~ 1 + x1 + x2 + x3 + x4,
                  data = df,
                  chains = 1, cores =1, iter=40000, thin=20)
prior<-prior_summary(post2)

# posterior sample
post.sample <- as.matrix(post2)

#no need for the intercept, last parameter is sigma
post.sample <- post.sample[,-1]

# plots

color_scheme_set("blue")

sample.plot.ex1 <- mcmc_areas(
  post.sample,
  prob = 0.95, # 95% credible intervals
  prob_outer = 1, # whole distribution
  point_est = "median"
)
sample.plot.ex1 <- sample.plot.ex1 +
       scale_y_discrete(labels = c(expression(beta*'1'),expression(beta*'2'),expression(beta*'3'), expression(beta*'4'), expression(sigma)))


# calculate R^2 of submodels post.sample[,5] represents the sigma parameter samples
df.rtwos <- allSubsetRtwos(df[,2:5], post.sample[,1:4], post.sample[,5])
df.rtwos <- data.frame(df.rtwos)

# option to display the resulting R^2 values of all sub-models for each posterior sample
color_scheme_set("red")
df.rtwos.t <- t(df.rtwos)
r2plot.ex1 <- mcmc_parcoord(df.rtwos.t) #plot from bayesplot package
r2plot.ex1 <- r2plot.ex1 + scale_y_continuous(breaks=seq(0,1,0.1), limits=c(0,1))+ ylab(expression( ~ R^2)) +
 xlab('Model containing subset of predictors')+ theme(axis.text.x = element_text(angle=90, vjust = 1))+
       theme(axis.title.y = element_text(margin = margin(t = 0, r = 20, b = 0, l = 0)))


# Prepare matrix
LMG.Vals.I<-matrix(0, 4, dim(df.rtwos)[2]) #LMG values (Independent component)
LMG.Vals.J<-matrix(0, 4, dim(df.rtwos)[2]) #Joint contributions
LMG.Vals.T<-matrix(0, 4, dim(df.rtwos)[2]) # Total

# Calculate the LMG and joint contribution values for each posterior joint parameter sample
for(i in 1:dim(df.rtwos)[2]){

  gofn<-df.rtwos[,i]

  #LMG calculation (needs as input dataframe with R^2 values of all sub-models)
  obj.Gelman<-partition(gofn, pcan = 4, var.names = names(df[,2:5]))

  LMG.Vals.I[,i]=obj.Gelman$IJ[,1]
  LMG.Vals.J[,i]=obj.Gelman$IJ[,2]
       LMG.Vals.T[,i]=obj.Gelman$IJ[,3]
}
```

```r
varinames <- row.names(obj.Gelman$IJ)


# posterior LMG distribution of each variable
quantile(LMG.Vals.I[1,], c(0.025, 0.5, 0.975))
quantile(LMG.Vals.I[2,], c(0.025, 0.5, 0.975))
quantile(LMG.Vals.I[3,], c(0.025, 0.5, 0.975))
quantile(LMG.Vals.I[4,], c(0.025, 0.5, 0.975))


# some example how the LMG distributions could be displayed
dat <- data.frame(t(LMG.Vals.I))

pairs.chart <- ggpairs(dat, lower = list(continuous = "density"), upper = list(continuous = "cor")) +
        ggplot2::theme(axis.text = element_text(size = 6))
```

R-Code A.4: LMG calculations assuming stochastic predictors (Bayesian covariance estimation in JAGS)

```r
#------------------------------------------------------------------------------
# In the following example we know that the predictors are coming from a normal distribution.
# The covariance matrix of the predictors can therefore be estimated in a Bayesian way.
# The package JAGS is used. The uncertainty of the predictors can therefore be included
# in the LMG calculations.
# Code adopted from http://doingbayesiandataanalysis.blogspot.com/2017/06/bayesian-estimation-of-correlations-and.html
#------------------------------------------------------------------------------


# Assemble data for sending to JAGS:
zy = df[,2:5]

dataList = list(
        zy = zy , # data
        Ntotal =  nrow(zy) , # number of individual observations
        Nvar = ncol(zy) , # number of timepoints
        zRscal = ncol(zy) ,  # scale for Wishart prior
        zRmat = diag(x=1,nrow=ncol(zy))  # identity matrix for Wishart prior
)

# Define the model:

# likelihood multivariate normal distribution,
# Wishart prior for covariance matrix (dwish)
# Normal distribution for means

modelString = "
model {
for ( i in 1:Ntotal ) {
zy[i,1:Nvar] ~ dmnorm( zMu[1:Nvar] , zInvCovMat[1:Nvar,1:Nvar] )
}
for ( varIdx in 1:Nvar ) { zMu[varIdx] ~ dnorm( 0 , 1/2^2 ) }
zInvCovMat ~ dwish( zRmat[1:Nvar,1:Nvar] , zRscal )
# Convert invCovMat to sd and correlation:
zCovMat <- inverse( zInvCovMat )

}
" # close quote for modelString
writeLines( modelString , con="Jags-MultivariateNormal-model.txt" )

# Run the chains:
nChain = 3
nAdapt = 500
nBurnIn = 500
nThin = 10
nStepToSave = 20000

# run the model
jagsModel = jags.model( file="Jags-MultivariateNormal-model.txt", data=dataList , n.chains=nChain, n.adapt=nAdapt)
update( jagsModel , n.iter=nBurnIn )
codaSamples = coda.samples( jagsModel , variable.names=c('zCovMat'), n.iter=nStepToSave/nChain*nThin , thin=nThin )

parameterNames = varnames(codaSamples) # get all parameter names

# Posterior distribution of predictor covariance matrix
mcmcMat = as.matrix(codaSamples)
chainLength = nrow(mcmcMat)
covMat <- array(NA, c(4,4,chainLength))

# reshape covariance matrix samples
for (i in 1:chainLength){
covMat[1:4,1:4,i]<-matrix(mcmcMat[i,], 4, 4) # covariance matrix sample for each i
}

#random sample from the distribution, no time for all samples (see next step)
covMat <- covMat[1:4,1:4,sample(1:20000, replace=F)]

n.boot = 1000 # number of covariance samples that we draw for the LMG calculations
covMat <- covMat[,,1:n.boot]

# use allSubsetRtwos() function to calculate explained variance of all sub-models for each posterior joint parameter sample
# for each posterior covariance sample
df.rtwos.covm <- allSubsetRtwos(df[,2:5], post.sample[,1:4], post.sample[,5], covMat)
df.rtwos.covm <- data.frame(df.rtwos.covm)
```

```
# use hier.part package to calculate the LMG values for each posterior joint parameter sample
LMG.Vals.I.covm <- foreach(i = 1:dim(df.rtwos.covm)[2], .combine = cbind,  .packages = c('hier.part')) %dopar%{

        gofn<-df.rtwos.covm[,i]

        partition(gofn, pcan = 4, var.names = names(df[,2:5]))$IJ[,1]

}
```

R-Code A.5: LMG calculations assuming stochastic predictors with bootstrapped covariance matrix

```
# Code to calculate LMG values for stochastic predictors by using bootstrapped predictor covariance samples.

# for parallel computing
 myCluster <- makeCluster(7, # number of cores to use
                                                                                    type = "PSOCK")
 registerDoParallel(myCluster)

#--------------------------------------------------------------------------------


boot.M <- bootcov(df[,2:5], 1000) # non-parametric bootstrapping

# bootcov <- function(df, boot.n){
#  len <- nrow(df)
#  cov.m <- cov(df)
#  l <- dim(cov.m)[1]
#  M.boot <- array(NA, c(l,l,boot.n))
#  M.boot[,,1] <- cov(df)
#  for (i in 2 :boot.n){
#  dfs <- df[sample(1:len, replace=T),]
#  M.boot[,,i] <- cov(dfs)
#  }
#
#  return(M.boot)
# }


# Calculate LMG values for all sub-models for each posterior joint parameter sample for each predictor covariance sample
df.rtwos.boot <- allSubsetRtwos(df[,2:5], post.sample[,1:4], post.sample[,5], boot.M)
df.rtwos.boot <- data.frame(df.rtwos.boot)

# LMG calculations using the hier.part package
LMG.Vals.I.boot <- foreach(i = 1:dim(df.rtwos.boot)[2], .combine = cbind,  .packages = c('hier.part')) %dopar%{

        gofn<-df.rtwos.boot[,i]

        partition(gofn, pcan = 4, var.names = names(df[,2:5]))$IJ[,1]
}
```

A.2.2 Empirical data example

R-Code A.6: LMG calculations assuming non-stochastic predictors

```
# fit model in rstanarm using default priors
bayes.hs <- stan_glm(paragrap ~ . ,
                     data = hs.data,
                     chains = 1, cores = 1, iter=40000, thin=20) # This results in a posterior sample size of 1000

prior <- prior_summary(bayes.hs)

post.sample <- as.matrix(bayes.hs)

dt <- data.frame(post.sample[1:10,2:6]) #put in data frame for easier plotting

# no need for the intercept, last parameter is sigma
post.sample <- post.sample[,-1]


# Plots of posteriors
color_scheme_set("green")

sample.plot.empi <- mcmc_areas(
  post.sample,
  prob = 0.95, # 95% credible intervals
  prob_outer = 0.995, # 99%
  point_est = "median",
  pars=c("general",  "sentence", "wordc" ,"wordm")
)

sample.plot.empi <- sample.plot.empi +
        scale_y_discrete(labels = c(expression(beta*'1'),expression(beta*'2'),expression(beta*'3'), expression(beta*'4')))

# Calculate R2 of all sub-models from posterior joint parameter samples
df.rtwos <- allSubsetRtwos(hs.data[,2:5], post.sample[,1:4], post.sample[,5])
df.rtwos <- data.frame(df.rtwos)


# Plot of R2 values for all sub-models for each posterior joint parameter sample
color_scheme_set("purple")
df.rtwos.t <- t(df.rtwos)
r2plot.ex1 <- mcmc_parcoord(df.rtwos.t)
r2plot.ex1 <- r2plot.ex1 + scale_y_continuous(breaks=seq(0,1,0.1), limits=c(0,1))+ ylab(expression( ~ R^2)) +
        xlab('Model containing subset of predictors')+   theme(axis.text.x = element_text(angle=90, vjust = 1))+
        theme(axis.title.y = element_text(margin = margin(t = 0, r = 20, b = 0, l = 0)))

# Prepare matrix to fill with LMG (I), Joint Contribution (J), and Total (T) values
LMG.Vals.I<-matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.J<-matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.T<-matrix(0, 4, dim(df.rtwos)[2])

# Calculation of LMG values for each posterior joint parameter sample
for(i in 1:dim(df.rtwos)[2]){

  gofn<-df.rtwos[,i]

  obj.Gelman<-partition(gofn, pcan = 4, var.names = names(hs.data[,2:5]))

  LMG.Vals.I[,i]=obj.Gelman$IJ[,1]
  LMG.Vals.J[,i]=obj.Gelman$IJ[,2]
        LMG.Vals.T[,i]=obj.Gelman$IJ[,3]
}
```

A.3 Code used in chapter 4

R-Code A.7: Data generating code for random intercept model

```
# Simulate data for random intercept model

sub <- 1:20  # number of subjects
subi <- rnorm(20, 0, 4)  # random intercept variance = 4^2
subi <- rep(subi, 4)
t <- c(0, 1, 2, 3)
t <- c(rep(0, 20), rep(1, 20), rep(2, 20), rep(3, 20))  # 4 timepoints

mu <- rep(0, 4)

# Predictors are assumed to come from a multivariate normal distribution

# Predictor covariance matrix
sig <- matrix(0.4, 4, 4)
diag(sig) <- 1
sig[3, 4] <- 0.9
sig[4, 3] <- 0.9
sig[1, 2] <- 0.3
sig[2, 1] <- 0.3

# draw predictor values from multivariate normal distribution
rawvars <- mvrnorm(n = 80, mu = mu, Sigma = sig)

x1 <- t + rawvars[, 1]
x2 <- t + rawvars[, 2]
x3 <- t + rawvars[, 3]
x4 <- t + rawvars[, 4]

# regression parameters
b1 <- b2 <- 1
b3 <- b4 <- 2

y <- x1 * b1 + x2 * b2 + x3 * b3 + x4 * b4 + subi + rnorm(80, 0, 0.1)  # dependent variable
df <- data.frame(y = y, x1 = x1, x2 = x2, x3 = x3, x4 = x4, sub = rep(sub,
    4))
```

R-Code A.8: LMG calculations with focus on within-subject variances

```
# calculate LMG values with focus on within subject variance components

# fit random intercept model in rstanarm with default priors
fit <- stan_glmer(y ~ x1 + x2 + x3 + x4 + (1 | sub), data = df, chains = 4,
    cores = 4)

post.sample <- as.matrix(fit)

post.betas <- post.sample[, 2:5]  # the four regression parameters
post.sigmas <- post.sample[, (ncol(post.sample) - 1)]  # within subject error

# Calculate R^2 values for all sub-models for each posterior joint
# parameter sample
df.rtwos <- allSubsetRtwos(df[, 2:5], post.betas, post.sigmas)
df.rtwos <- data.frame(df.rtwos)


LMG.Vals.I <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.J <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.T <- matrix(0, 4, dim(df.rtwos)[2])

# Calculate LMG values for each joint parameter sample by using the
# hier.part package
for (i in 1:dim(df.rtwos)[2]) {
    gofn <- df.rtwos[, i]

    obj.Gelman <- partition(gofn, pcan = 4, var.names = names(df[, 2:5]))

    LMG.Vals.I[, i] = obj.Gelman$IJ[, 1]
    LMG.Vals.J[, i] = obj.Gelman$IJ[, 2]
    LMG.Vals.T[, i] = obj.Gelman$IJ[, 3]
}

varnames <- row.names(obj.Gelman$IJ)
```

R-Code A.9: LMG calculations with focus on total variance

```
# calculate LMG values with focus on total variance components

# post sigmas includes now within subject error and random intercept
# variance term
post.sigmas <- post.sample[, (ncol(post.sample) - 1):ncol(post.sample)]

# post betas same as in within-subject focus code

# Calculate R^2 values for all sub-models for each posterior joint
# parameter sample
df.rtwos <- allSubsetRtwos(df[, 2:5], post.betas, post.sigmas)
df.rtwos <- data.frame(df.rtwos)


LMG.Vals.I <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.J <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.T <- matrix(0, 4, dim(df.rtwos)[2])

# Calculate LMG values for each joint parameter sample by using the
# hier.part package
for (i in 1:dim(df.rtwos)[2]) {
    gofn <- df.rtwos[, i]

    obj.Gelman <- partition(gofn, pcan = 4, var.names = names(df[, 2:5]))

    LMG.Vals.I[, i] = obj.Gelman$IJ[, 1]
    LMG.Vals.J[, i] = obj.Gelman$IJ[, 2]
    LMG.Vals.T[, i] = obj.Gelman$IJ[, 3]
}
```

R-Code A.10: Data generation for marginal model

```
# data generation of marginal model

# Predictor generation Predictors assumed to come from a multivariate
# normal distribution
sub <- 1:20  # number of subjects
subi <- rnorm(20, 0, 1)
subi <- rep(subi, 4)
mu <- rep(0, 4)
sig <- matrix(0.4, 4, 4)  # covariance matrix of predictors
diag(sig) <- 1
sig[3, 4] <- 0.9
sig[4, 3] <- 0.9
sig[1, 2] <- 0.3
sig[2, 1] <- 0.3
rawvars <- mvrnorm(n = 80, mu = mu, Sigma = sig)
cov(rawvars)
t <- c(rep(1, 20), rep(2, 20), rep(3, 20), rep(4, 20))
x1 <- t + rawvars[, 1]
x2 <- t + rawvars[, 2]
x3 <- t + rawvars[, 3]
x4 <- t + rawvars[, 4]

# unstructured covariance matrix of error term per subject
Sig <- matrix(3, 4, 4)
diag(Sig) <- 10
u <- rep(0, 4)
Sig[1, 1] <- 5
Sig[2, 2] <- 7
Sig[3, 4] <- 8
Sig[4, 3] <- 8
Sig[1, 2] <- 4
Sig[2, 1] <- 4
error <- mvrnorm(20, u, Sig)

# regression parameters
b1 <- b2 <- 1
b3 <- b4 <- 2

# dependent variable
y <- x1 * b1 + x2 * b2 + x3 * b3 + x4 * b4 + c(error)
df <- data.frame(y = y, x1 = x1, x2 = x2, x3 = x3, x4 = x4, sub = rep(sub,
    4), t = t)

# Prepare for Bayesian framework

Y <- matrix(df[, "y"], 20, 4, byrow = F)
x1 <- matrix(df[, "x1"], 20, 4, byrow = F)
x2 <- matrix(df[, "x2"], 20, 4, byrow = F)
x3 <- matrix(df[, "x3"], 20, 4, byrow = F)
x4 <- matrix(df[, "x4"], 20, 4, byrow = F)

N = 20  # subjects
M = 4  # repeated measures
```

R-Code A.11: LMG calculations for the marginal model

```
#-------------------------------------------

# Inference about marginal model data with unstructured error covariance
# matrix likelihood: multivariate normal distribution with Wishart prior
# for error covariance matrix and Normal distribution prior for means

modelString <- "model{

# Likelihood
for(i in 1:N){
Y[i,1:M] ~ dmnorm(mu[i,1:M],Omega[1:M,1:M])
for(j in 1:M){
mu[i,j] <- beta0 + beta1*x1[i,j]+ beta2*x2[i,j]+ beta3*x3[i,j] + beta4*x4[i,j]
}}

# Priors

Omega[1:M, 1:M] ~dwish(zRmat[1:M,1:M] , zRscal)
Sigma[1:M, 1:M] <- inverse(Omega)

beta0      ~ dnorm(0,0.001)
beta1      ~ dnorm(0,0.001)
beta2      ~ dnorm(0,0.001)
beta3      ~ dnorm(0,0.001)
beta4      ~ dnorm(0,0.001)

}"

writeLines(modelString, con = "Jags-MultivariateNormal-model.txt")

# run model
model <- jags.model(textConnection(modelString), data = list(Y = Y, N = N,
    M = M, x1 = x1, x2 = x2, x3 = x3, x4 = x4, zRscal = ncol(Y), zRmat = diag(x = 1,
        nrow = ncol(Y))), n.chains = 3)

samp <- coda.samples(model, variable.names = c("beta1", "beta2", "beta3", "beta4",
    "Sigma"), n.iter = 20000, progress.bar = "none")


# posterior parameter distribution (interested in regression parameters and
# covariance matrix of the error term)

samp <- coda.samples(model, variable.names = c("beta1", "beta2", "beta3", "beta4",
    "Sigma[1,1]", "Sigma[2,2]", "Sigma[3,3]", "Sigma[4,4]"), n.iter = 20000,
    thin = 20, progress.bar = "none")

post.betas <- as.matrix(samp[[1]][, 5:8])  # regression parameters
post.sigmas <- as.matrix(samp[[1]][, 1:4])  # diagonal elements of covariance matrix

# only need mean of post.sigmas per joint parameter sample
post.sigmas.mean <- apply(post.sigmas, 1, mean)

# Calculate R^2 values for each posterior joint parameter sample
df.rtwos <- allSubsetRtwos(df[, 2:5], post.betas, post.sigmas.mean)
df.rtwos <- data.frame(df.rtwos)

# Calculate LMG values

# Prepare matrix to fill with LMG (I), Joint contribution (J) and Total (T)
# values
LMG.Vals.I <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.J <- matrix(0, 4, dim(df.rtwos)[2])
LMG.Vals.T <- matrix(0, 4, dim(df.rtwos)[2])

# Calculate LMG (I), Joint contribution (J) and Total (T) values for each
# joint posterior parameter sample
for (i in 1:dim(df.rtwos)[2]) {
    gofn <- df.rtwos[, i]
    obj.Gelman <- partition(gofn, pcan = 4, var.names = names(df[, 2:5]))
    LMG.Vals.I[, i] = obj.Gelman$IJ[, 1]
    LMG.Vals.J[, i] = obj.Gelman$IJ[, 2]
    LMG.Vals.T[, i] = obj.Gelman$IJ[, 3]
}
```

A.4 Software

Figure 3.1, Figure 3.2, Figure 3.5, and Figure 3.6 of chapter 3 were plotted with the `bayesplot` package (Gabry, 2017). Figure 3.3 and Figure 3.7 of chapter 3 were plotted with the `GGally` package which is an extension to the `ggplot2` (Wickham, 2016) package.

```
sessionInfo()
```

```
## R version 3.4.3 (2017-11-30)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.6
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
## [1] de_CH.UTF-8/de_CH.UTF-8/de_CH.UTF-8/C/de_CH.UTF-8/de_CH.UTF-8
##
## attached base packages:
## [1] parallel  grid      stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] bindrcpp_0.2       bayesplot_1.6.0    brelimp_0.0.0.9000
##  [4] doParallel_1.0.11  iterators_1.0.10   foreach_1.4.4
##  [7] biostatUZH_1.8.0   MBESS_4.4.3        kableExtra_0.9.0
## [10] xtable_1.8-2       stargazer_5.2.2    brinla_0.1.0
## [13] INLA_17.06.20      sp_1.2-7           corpcor_1.6.9
## [16] rjags_4-6          coda_0.19-1        relaimpo_2.2-2
## [19] mitools_2.3        survey_3.32-1      survival_2.41-3
## [22] Matrix_1.2-12      boot_1.3-20        MASS_7.3-47
## [25] GGally_1.4.0       ggplot2_2.2.1      rstanarm_2.17.2
## [28] Rcpp_0.12.18       hier.part_1.0-4    gtools_3.5.0
## [31] knitr_1.19
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-131       matrixStats_0.53.1 xts_0.10-1
##  [4] RColorBrewer_1.1-2 threejs_0.3.1      httr_1.3.1
##  [7] rprojroot_1.3-2    rstan_2.17.3       tools_3.4.3
## [10] backports_1.1.2    R6_2.2.2           DT_0.4
## [13] lazyeval_0.2.0     colorspace_1.3-2   gridExtra_2.3
## [16] compiler_3.4.3     rvest_0.3.2        formatR_1.5
## [19] xml2_1.2.0         shinyjs_1.0        labeling_0.3
## [22] colourpicker_1.0   scales_0.5.0       dygraphs_1.1.1.4
## [25] readr_1.1.1        ggridges_0.5.0     stringr_1.3.1
## [28] digest_0.6.15      StanHeaders_2.17.2 minqa_1.2.4
## [31] rmarkdown_1.8      base64enc_0.1-3    pkgconfig_2.0.1
## [34] htmltools_0.3.6    lme4_1.1-15        highr_0.6
## [37] htmlwidgets_1.0    rlang_0.2.1        rstudioapi_0.7
## [40] shiny_1.0.5        bindr_0.1          zoo_1.8-0
## [43] crosstalk_1.0.0    dplyr_0.7.4        inline_0.3.14
## [46] magrittr_1.5       loo_1.1.0          munsell_0.4.3
## [49] stringi_1.2.4      plyr_1.8.4         miniUI_0.1.1
## [52] lattice_0.20-35    splines_3.4.3      hms_0.3
## [55] pillar_1.2.3       igraph_1.1.2       markdown_0.8
## [58] shinystan_2.4.0    reshape2_1.4.2     codetools_0.2-15
## [61] stats4_3.4.3       rstantools_1.4.0   glue_1.3.0
## [64] evaluate_0.10.1    nloptr_1.0.4       httpuv_1.3.5
## [67] gtable_0.2.0       reshape_0.8.7      assertthat_0.2.0
## [70] mime_0.5           rsconnect_0.8.8    viridisLite_0.2.0
## [73] tibble_1.4.2       shinythemes_1.1.1
```

Bibliography

Alexander, D. L., Tropsha, A., and Winkler, D. A. (2015). Beware of R^2: Simple, Unambiguous Assessment of the Prediction Accuracy of QSAR and QSPR Models. *Journal of Chemical Information and Modeling*, **55**, 1316–1322.

Bland, J. M. and Altman, D. G. (1995). Calculating correlation coefficients with repeated observations: Part 1–Correlation within subjects. *British Medical Journal*, **310**, 446.

Chevan, A. and Sutherland, M. (1991). Hierarchical partitioning. *American Statistician*, **45**, 90–96.

Fitzmaurice, G. M., Laird, N. M., and Ware, J. H. (2011). *Applied Longitudinal Analysis*. Wiley.

Gabry, J. (2017). bayesplot: Plotting for Bayesian models. R package version 1.6.0.

Gelman, A., Goodrich, B., Gabry, J., and Ali, I. (2017). R-squared for Bayesian regression models. Technical report.

Grömping, U. (2006). Relative Importance for Linear Regression in R : The Package relaimpo. *Journal of Statistical Software*, **17**, 1–27.

Grömping, U. (2007). Estimators of relative importance in linear regression based on variance decomposition. *American Statistician*, **61**, 139–147.

Grömping, U. (2015). Variable importance in regression models. *Wiley Interdisciplinary Reviews: Computational Statistics*, **7**, 137–152.

Holzinger, K. J. and Swineford, F. (1939). A study in factor analysis: The stability of a bi-factor solution. *Supplementary Educational Monographs*, **48**, 1–91.

Jackman, S. (2009). *Bayesian Analysis for the Social Sciences*. Wiley.

Kelley, K. (2017). Mbess (version 4.0.0 and higher) [computer software and manual].

Kvalseth, T. O. (1985). Cautionary Note about R^2. *The American Statistician*, **39**, 279.

Link, W. A. and Eaton, M. J. (2012). On thinning of chains in MCMC. *Methods in Ecology and Evolution*, **3**, 112–115.

Nakagawa, S. and Schielzeth, H. (2013). A general and simple method for obtaining R^2 from generalized linear mixed-effects models. *Methods in Ecology and Evolution*, **4**, 133–142.

Nimon, K., Lewis, M., Kane, R., and Haynes, R. M. (2008). An R package to compute commonality coefficients in the multiple regression case: An introduction to the package and a practical example. *Behavior Research Methods*, **40**, 457–466.

Nimon, K. F. and Oswald, F. L. (2013). Understanding the Results of Multiple Linear Regression: Beyond Standardized Regression Coefficients. *Organizational Research Methods*, **16**, 650–674.

Plummer, M. (2017). *JAGS Version 4.3.0 user manual*.

Schäfer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, **4**, 1–30.

Stan Development Team (2016). rstanarm: Bayesian applied regression modeling via Stan. R package version 2.13.1.

Stan Development Team (2017). *Stan Modeling Language: User's Guide and Reference Manual*.

van de Schoot, R., Winter, S. D., Ryan, O., Zondervan-Zwijnenburg, M., and Depaoli, S. (2017). A systematic review of Bayesian articles in psychology: The last 25 years. *Psychological Methods*, **22**, 217–239.

Walsh, C. and Nally, R. M. (2015). Hierarchical Partitioning. Technical report.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.