ON LASSO ESTIMATION OF LINEAR MIXED MODEL FOR HIGH DIMENSIONAL LONGITUDINAL DATA

A Dissertation
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Qian Wen

In Partial Fulfillment of the Requirements
for the Degree of
DOCTOR OF PHILOSOPHY

Major Department:
Statistics

June 2021

Fargo, North Dakota

NORTH DAKOTA STATE UNIVERSITY

Graduate School

Title

ON LASSO ESTIMATION OF LINEAR MIXED MODEL FOR HIGH DIMENSIONAL LONGITUDINAL DATA

By

Qian Wen

The supervisory committee certifies that this dissertation complies with North Dakota State University's regulations and meets the accepted standards for the degree of

DOCTOR OF PHILOSOPHY

SUPERVISORY COMMITTEE:

Prof. Gang Shen
Chair

Mr. Curt Doetkott

Prof. Megan Orr

Prof. Changhui Yan

Approved:

06/22/2021
Date

Rhonda Magel
Department Chair

ABSTRACT

With the advancement of technology in data collection, repeated measurements with high dimensional covariates have become increasingly common. The classical statistics approach for modeling the data of this kind is via the linear mixed model with temporally correlated error. However, most of the research reported in the literature for variable selection is for independent response data. In this study, the proposed algorithm employs Expectation and Maximization (EM) and Least Absolute Shrinkage and Selection Operator (LASSO) approaches under the linear mixed model scheme with the assumption of Gaussianity, an approach that works for data with interdependence. Our algorithm involves two steps: 1.Variance-covariance components estimation by EM; and 2.Variable selection by LASSO. The crucial challenge arises from the fact that linear mixed models usually allow structured variance-covariance, which, in return, renders complexity in its estimation: No explicit maxima in general in the M-step of the EM algorithm. Our EM algorithm uses one iteration of projection gradient descent method, which turns out to be quite computationally efficient compared with the classical EM algorithm because it obviates the process of finding the maxima of the variance-covariance components in the M-step. With the estimates of variance-covariance components obtained from step 1, the LASSO estimation is executed on the full log-likelihood function imposed with an $l1$ regularization. The LASSO method has the effect of shrinking all coefficients towards zero, which plays a variable selection role. We apply the gradient descent algorithm to find LASSO estimates and the pathwise coordinate descent to set up the tuning parameter for the penalized log-likelihood function. The simulation studies are carried out under the assumption that measurement errors of each subject are of first-order autoregressive AR(1) correlation structure. The numerical results show that the variance-covariance parameters estimates by our method are comparable to the classic Newton-Raphson (NR) method in the simple case and outperforms NR method when the variance-covariance matrix having a complex structure. Moreover, our method successfully identifies all the relevant explanatory variables and most of the redundant explanatory variables. The proposed method is also applied to a life data and the result is very reasonable.

ACKNOWLEDGEMENTS

Foremost, I would like to express my deepest gratitude to my advisor Dr. Gang Shen for his guidance, support, and patience throughout my PhD journey. He is the best advisor I could ever ask for.

In addition to my advisor, I would also like to thank the rest of my committee, Mr. Curt Doetkott, Dr. Megan Orr, and Dr. Changhui Yan, for their time and feedback.

Additionally, my sincere thanks goes to Mr. Curt Doetkott for the opportunity of working at Statistical Consulting Center. The skills I learned and the experiences I had there will be treasured forever.

I thank Kristina Caton at the Center for Writers for checking my grammar, which greatly improved my manuscript.

I thank Dr. Xinhua Jia and her student Uday Bhanu Prakash Vaddevolu, from department of Agricultural and Biosystems Engineering, for providing me the life data.

Last but not least, I would like to thank my family, my husband, my son, and my parents, for their continued support and endless love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

1. INTRODUCTION

In the longitudinal study, measurements are repeatedly collected from the same subjects for a certain period of time; therefore, the data are usually correlated. Longitudinal data have been widely observed in areas such as the medical and health sciences and the social sciences. With the advancement of data collection and storage technology in the age of big data, longitudinal data of high dimension becomes more and more accessible. A typical way to deal with longitudinal data in statistics is via the linear mixed model (LMM), which, in its systematic component, allows both deterministic and random components. The random components could explain the additional source of variation in measurements due to heterogeneity of subjects that could not be explained by the covariates of subjects considered in the deterministic components.

There are two main tasks when analyzing longitudinal data of high dimension: 1. Modeling the correlation among the repeated measurements collected from same subject; and 2. Variable selection. Taking account of the correlation among the repeated measurements reduces the variability of the estimate due to time change within subject, while ignoring the correlation leads to misleading inferences about the parameters. Variable selection aims to build a meaningful model with as few covariates as possible since not all of the covariates collected contribute to explaining the response variable. In this work, we propose a new approach to accomplish those two tasks in two steps.

The estimation in LMM involves estimating the fixed effects and estimating the variance-covariance parameters. The preferred method used to estimate variance-covariance components is restricted maximum likelihood (REML) estimation [16]. Compared to Maximum Likelihood Estimation(MLE), REML yields estimates with smaller variance when the number of covariates is larger [9]. With the implementation of the REML technique, the likelihood function of LMM can be written as a product of marginal likelihood function and conditional likelihood function. The conditional likelihood function depends only on fixed effect and does not contain information about variance-covariance components [9]. The marginal likelihood function known as REML is used to estimate variance-covariance components. The REML estimators of the variance-covariance components are not only invariant to the fixed effects of the model, but are also free of the estimates

of the fixed effects [3]. To make full use of REML's strength and increase the computational efficiency, we propose a model with two steps: 1. estimate variance-covariance components with the REML; and 2. select and estimate the fixed effects with the full likelihood function.

The log-likelihood function of REML is not a linear function of variance-covariance parameters; therefore, an iterative algorithm is needed. In this study, the EM algorithm is chosen to iteratively maximize the log-likelihood function. However, there is a crucial challenge to performing the classic EM algorithm with longitudinal data: there is no closed form solution in M-step, i.e, another iterative scheme is required in each M-step. In order to increase the computational efficiency, we propose to use an EM algorithm with the M-step solved by one iteration of gradient descent algorithm. This concludes step 1 of our algorithm.

Step 2 of our algorithm involves using the least absolute shrinkage and select operator (LASSO) [19] to select the important explanatory variables. With the estimated variance-covariance components from step 1, the full log-likelihood function is equivalent to a quadratic loss function; therefore, the penalized full log-likelihood is equivalent to a penalized quadratic loss function of the fixed effect coefficients. Lasso minimizes penalized quadratic loss function by imposing an $l1$ regularization. With the $l1$ regularization, parameters are shrunk toward zero and some are exactly zero. Thus, implementing LASSO yields a parsimonious and meaningful model because most of the redundant explanatory variables can be removed.

The rest of this work is organized as follows: A review of the related work is presented in Section 2 and technical details of the proposed method are given in Section 3. Section 4 illustrates the performance of the proposed method with simulation study and real life data application. Section 5 includes conclusions and future directions.

2. LITERATURE REVIEW

As explained in Diggle [4], the variations of longitudinal data consist of variation between subjects, variation between times within each subject, and variation due to measurement errors. The Linear Mixed Model (LMM) is widely used to analyze the longitudinal data because it can incorporate these three sources of variation in the modeling procedure. The variance-covariance parameters of LMM are usually estimated by the restricted maximum likelihood (REML) procedures with the second order algorithms, such as the Newton Ralphson (NR) algorithm. However, the NR algorithm has two drawbacks: it becomes more computationally expensive as the number of parameters increases and it may fail to converge when the second derivatives of the objective function are complicated. Consequently, the Expectation and Maximization (EM) algorithm is a good alternative to the NR algorithm because the EM algorithm allows us to separately estimate the parameters involved in the random effects and the parameters corresponding to the temporally correlated error term. Again however, there is a problem in the use of the EM algorithm to estimate correlation parameters that pertain to time processes: no closed form solution exists; thus, an iterative solution is required at the M step. Therefore, Rai and Matthews [17] and Lange [12] suggested solving the M step by involving only one cycle of iterative solution of the maximization problem while keep the E-step unchanged. Like the classic EM algorithm, this modified version was shown to be self-consistent under suitable conditions [17]. In contrast to Foulley [8], who solved the M-step with one iteration of the NR algorithm, we propose to solve the M-step with one iteration of gradient descent algorithm, which is a first order algorithm.

The history of variable selection can be traced back to the 1960s when Beale, Kendall,and Mann [1] proposed the best subset selection and Efroymson [5] proposed the forward/backward stepwise selection. Even though these traditional variable selection methods seem easy to implement, they are very time consuming when the number of explanatory variables is large. Additionally, Breiman [2] claimed that these methods tend to lack stability due to their inherent discreteness. Modern variable selection techniques include the least absolute shrinkage and select operator (LASSO) technique proposed by Tibshirani [19] and the smoothly clipped absolute deviation (SCAD) method proposed by Fan [6]. Recently, there have been many research studies on

variable selection based on penalized log-likelihood and the computational issues of solving these problems within the linear model scheme, for example, Ni et al.[15], Fan and Li [7], and Lin et al. [13]. However, most of these works focused on select fixed effect and random effect either simultaneously or separately, and all of them assume residuals are independently identically distributed, which is unrealistic in the setting of longitudinal data. A study that is close to our work is presented by Lan [11]. Lan proposed selecting variables by the penalized likelihood estimation procedure with the SCAD penalty and estimating variance-covariance components with REML. However, she neglected the fact that REML estimations of variance-covariance parameters are invariant to fixed effects. As a result, the algorithm she proposed had some unnecessary steps.

A common concern for all the penalized variable selection techniques is the determination of the penalty parameter. The Akaike information criterion (AIC) and the Bayesian information criterion (BIC) are two popular criteria used in variable selection and model selection. Zou et al. [21] evaluated the performance of the AIC and BIC in the application of variable selection and concluded that BIC is better than AIC in the sense of identifying the sparsity of a model.

3. METHODOLOGY

3.1. Linear Mixed Model for Longitudinal Data

Longitudinal data set contains measurement taken repeatedly over time on a set of subjects, the Linear Mixed Model (LMM) for the subject i is:

$$\tilde{y}_i = X_i\tilde{\beta} + Z_i\tilde{u} + \tilde{e}_i \tag{3.1}$$

with i=1,2,...,m subjects, measurements of n_i time points are collected for subject i, $\tilde{y}_i$ is $n_i \times 1$ vector of observations taken from subject i over time, X_i is $n_i \times p$ design matrix for fixed effect for subject i, $\tilde{\beta}$ is $p \times 1$ vector of fixed effect parameters. The random effect $\tilde{u}'$ consists of q subvectors such that $\tilde{u}' = (\tilde{u}_1', ..., \tilde{u}_q')$ where $\tilde{u}_s$ is the vector of the sth random effect and is of length $b_s \times 1$. Let $b = \sum b_s$, then $\tilde{u}$ is $b \times 1$ vector and follows $N(0, G)$; Z_i is $n_i \times b$ design matrix corresponding to random effects for subject i; $\tilde{e}_i$ is stationary residual vector of i_{th} subject and follows $N\left(0, \sigma^2\Sigma_i\right)$, Σ_i depends on i through its dimension only. $\tilde{u}$ is independent of $\tilde{e}_i$ for any $i = 1, 2, 3, .., m$. Stacking X_i, Z_i and $\tilde{e}_i$ respectively and setting $\Sigma = diag(\Sigma_1, \Sigma_2, ..., \Sigma_m)$, the linear mixed model in equation 3.1 becomes

$$\tilde{y} = X\tilde{\beta} + Z\tilde{u} + \tilde{e} \tag{3.2}$$

and the marginal distribution of $\tilde{y}$ is

$$\tilde{y} \sim \mathrm{N}(X\tilde{\beta}, H) \tag{3.3}$$

where $H = ZGZ' + \sigma^2\Sigma$. Denote $G = G(\tilde{\gamma})$ and $\Sigma = \Sigma(\tilde{\phi})$, where $\tilde{\gamma}$ and $\tilde{\phi}$ are vectors of variance parameters associated with random effects and error term respectively. Then we can define the variance parameter space as $\tilde{\xi} = (\sigma^2, \tilde{\gamma}, \tilde{\phi})$.

In general, the variance-covariance components in LMM are estimated by maximum likelihood, however maximum likelihood estimates of variance-covariance components are known as downward biased because it fails to take into account estimation of fixed effects; the bias is especially severe when the number of covariates p is large. To correct this bias, REML[16] is used to estimate variance-covariance components. Additionally, REML estimators are invariant to the fixed effects of the model, therefore, it is also free of the estimates of the fixed effects[9].

As proposed by Verbyla [20], the implementation of REML technique begins by considering nonsingular matrix $L = [L_1, L_2]$ where L_1 and L_2 are n $\times$ p and n $\times$ $(n-p)$ matrices respectively, and L_1 and L_2 satisfy $L_1'X = I_p$ and $L_2'X = 0$, then transform $\tilde{y}$ by left multiplying L', i.e.

$$L'\tilde{y} = \begin{bmatrix} L_1'\tilde{y} L_2'\tilde{y} \end{bmatrix} = \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{bmatrix} \tag{3.4}$$

And the transformed data follows following distribution:

$$\begin{pmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{pmatrix} \sim N\left\{ \begin{pmatrix} \tilde{\beta} \\ 0 \end{pmatrix}, \begin{pmatrix} L_1'HL_1 & L_1'HL_2 \\ L_2'HL_1 & L_2'HL_2 \end{pmatrix} \right\} \tag{3.5}$$

we can show that for L_2 of such choice and any positive definite matrix A following equality holds

$$L_2(L_2'AL_2)^{-1}L_2' = A^{-1} - A^{-1}X(X'A^{-1}X)^{-1}X'A^{-1} \tag{3.6}$$

Lemma 1. *Given L_1 and L_2 described as above, for any $n \times n$ positive defined matrix A, we have*

$$L_2(L_2'AL_2)^{-1}L_2' = A^{-1} - A^{-1}X(X'A^{-1}X)^{-1}X'A^{-1}$$

Proof. Let $C(B)$ be the linear space spanned by the column vectors of matrix B of dimension $n \times p$ (with $rank(B) = p < n$). Note that for any non-singular matrix Q of dimension $p \times p$, $C(BQ) = C(B)$ and $(BQ)[(BQ)'(BQ)]^{-1}(BQ)' = B(B'B)^{-1}B'$, uniqueness of the projection matrix $B(B'B)^{-1}B'$ follows.

Note with $C(B_1) \cup C(B_2) = R^n$ and $C(B_1) \cap C(B_2) = \tilde{0}$, $\forall \tilde{y} \in R^n$ has a unique decomposition, i.e., $\tilde{y} = \tilde{y}_1 + \tilde{y}_2$ where $\tilde{y}_1 \in C(B_1)$, $\tilde{y}_2 \in C(B_2)$. Note that $\forall \tilde{y} \in R^n$,

$$\{B_2(B_2'B_2)^{-1}B_2'\}\tilde{y} = \{B_2(B_2'B_2)^{-1}B_2'\}(\tilde{y}_1 + \tilde{y}_2) = \{B_2(B_2'B_2)^{-1}B_2'\}\tilde{y}_2 = \tilde{y}_2,$$

$$\{I_n - B_1(B_1'B_1)^{-1}B_1'\}\tilde{y} = \tilde{y} - B_1(B_1'B_1)^{-1}B_1'\tilde{y} = \tilde{y} - \tilde{y}_1 = \tilde{y}_2,$$

$B_2(B_2'B_2)^{-1}B_2' = I_n - B_1(B_1'B_1)^{-1}B_1'$ concludes.

Now, for any positive definite matrix A of dimension $n \times n$, let $B_1 = A^{-1/2}X$ and $B_2 = A^{1/2}L_2$. Clearly, $C(B_1) \cup C(B_2) = R^n$ and $C(B_1) \cap C(B_2) = \tilde{0}$, since $B_1'B_2 = X'L_2 = \tilde{0}$. Apply the equation obtained above, then

$$A^{1/2}L_2(L_2'AL_2)^{-1}L_2'A^{1/2} = I_n - A^{-1/2}X(X'A^{-1}X)^{-1}X'A^{-1/2}.$$

Multiplying $A^{-1/2}$ to both the left and right end of the two sides of the equation, the conclusion follows. □

The probability density function of transformed data $L'\tilde{y}$ can be written as the product of conditional density function of $\tilde{y}_1$ given $\tilde{y}_2$ and the marginal density function of $\tilde{y}_2$. The conditional distribution of $\tilde{y}_1$ given $\tilde{y}_2$ is:

$$\tilde{y}_1|\tilde{y}_2 \sim N(\tilde{\beta} + L_1'HL_2(L_2'HL_2)^{-1}\tilde{y}_2, (X'H^{-1}X)^{-1}) \tag{3.7}$$

where

$$\begin{aligned} var(\tilde{y}_1|\tilde{y}_2) &= L_1'HL_1 - L_1'HL_2(L_2'HL_2)^{-1}L_2'HL_1 \\ &= L_1'(H - HL_2(L_2'HL_2)^{-1}L_2'H)L_1 \quad \text{by lemma1} \\ &= L_1'X(X'H^{-1}X)^{-1}X'L_1 \\ &= (X'H^{-1}X)^{-1} \end{aligned}$$

and the conditional log-likelihood function $l(\tilde{\xi}; \tilde{y}_1|\tilde{y}_2)$ excluding constant term can be written as

$$l(\tilde{\beta}, \tilde{\xi}; \tilde{y}_1|\tilde{y}_2) = -\frac{1}{2}[-log|X'H^{-1}X| + (\tilde{y}_1 - \tilde{\beta} - \tilde{y}_2^*)'(X'H^{-1}X)(\tilde{y}_1 - \tilde{\beta} - \tilde{y}_2^*)]. \tag{3.8}$$

where $\tilde{y}_2^* = L_1'HL_2(L_2'HL_2)^{-1}\tilde{y}_2$. The marginal distribution of $\tilde{y}_2$ is:

$$\tilde{y}_2 \sim N(\tilde{0}, L_2'HL_2) \tag{3.9}$$

and corresponding log-likelihood function (excluding constant term) is

$$l(\tilde{\xi}; \tilde{y}_2) = -\frac{1}{2}[log|L_2'HL_2| + \tilde{y}_2'(L_2'HL_2)^{-1}\tilde{y}_2. \tag{3.10}$$

Then the the full log-likelihood function of $L'y$ excluding constant term is:

$$\begin{aligned} l(\tilde{\beta}, \tilde{\xi}; L'\tilde{y}) &= l(\tilde{\xi}; \tilde{y}_2) + l(\tilde{\beta}, \tilde{\xi}; \tilde{y}_1|\tilde{y}_2) \\ &= -\frac{1}{2}[log|L_2'HL_2| + \tilde{y}_2'(L_2'HL_2)^{-1}\tilde{y}_2 \\ &\quad -\frac{1}{2}[-log|X'H^{-1}X| + (\tilde{y}_1 - \tilde{\beta} - \tilde{y}_2^*)'(X'H^{-1}X)(\tilde{y}_1 - \tilde{\beta} - \tilde{y}_2^*)] \end{aligned} \tag{3.11}$$

Usually, the marginal log-likelihood function $l(\tilde{\xi}; y_2)$ is used to estimate variance-covariance parameters, and Sprott [18] claimed that y_2 is marginally sufficient for $\tilde{\xi}$. Fixed effects are estimated by maximizing the conditional log-likelihood function $l(\tilde{\beta}, \tilde{\xi}; \tilde{y}_1|\tilde{y}_2)$ with the estimated variance-covariance parameters. Even the expression of $l(\tilde{\beta}, \tilde{\xi}; \tilde{y}_1|\tilde{y}_2)$ contains unknown parameter $\tilde{\xi}$, Patterson [16] stated that, in the absence of outside knowledge of $\tilde{\beta}$, y_1 has no information about variance-covariance parameter $\tilde{\xi}$.

3.2. Variance Estimation by EM algorithm

The log-likelihood function of marginal distribution of $\tilde{y}_2$ is used for REML estimation of variance parameters. In order to use EM algorithm, complete data set is defined as $\tilde{y}_c = (\tilde{y}_2', \tilde{u})$. Note that the parameter space of REML log-likelihood is defined as $\tilde{\xi} = (\sigma^2, \tilde{\gamma}, \tilde{\phi})$. The joint distribution of $\tilde{y}_2$ and $\tilde{u}$ is

$$\begin{pmatrix} \tilde{y}_2 \\ \tilde{u} \end{pmatrix} \sim N\left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} L_2'HL_2 & L_2'ZG \\ GZ'L_2 & G \end{pmatrix} \right\} \tag{3.12}$$

And the conditional distribution of $\tilde{y}_2$ given $\tilde{u}$ and marginal distribution of $\tilde{u}$ are

$$\tilde{y}_2|\tilde{u} \sim N\left(L_2'Z\tilde{u}, \sigma^2 L_2'\Sigma L_2\right) \tag{3.13}$$

$$\tilde{u} \sim N(0, G) \tag{3.14}$$

Therefore the complete log-likelihood function is

$$\begin{aligned} l &= log\left(f\left(\tilde{y}_2, \tilde{u}\right)\right) \\ &= log\left(f\left(\tilde{y}_2|\tilde{u}\right)\right) + log\left(f\left(\tilde{u}\right)\right) \end{aligned} \tag{3.15}$$

Let $l_1 = log\left(f\left(\tilde{y}_2|\tilde{u}\right)\right)$ and $l_2 = log\left(f\left(\tilde{u}\right)\right)$, then excluding constant term

$$\begin{aligned} l_1 &= -\frac{1}{2}[log|\sigma^2 L_2'\Sigma L_2| + \sigma^{-2}\left(\tilde{y}_2 - L_2'Z\tilde{u}\right)'\left(L_2'\sigma L_2\right)^{-1}\left(\tilde{y}_2 - L_2'Z\tilde{u}\right)] \\ &= -\frac{1}{2}[log|\sigma^2 L_2'\Sigma L_2| + \sigma^{-2}\left(\tilde{y} - Z\tilde{u}\right)' L_2\left(L_2'\Sigma L_2\right)^{-1} L_2'\left(\tilde{y} - Z\tilde{u}\right)] \\ &= -\frac{1}{2}[log|\sigma^2 L_2'\Sigma L_2| + \left(\tilde{y} - Z\tilde{u}\right)' S\left(\tilde{y} - Z\tilde{u}\right)] \end{aligned} \tag{3.16}$$

where $S = \sigma^{-2}L_2\left(L_2'\Sigma L_2\right)^{-1}L_2' = \sigma^{-2}\Sigma^{-1} - \sigma^{-2}\Sigma^{-1}X(X'\Sigma^{-1}X)^{-1}X'\Sigma^{-1}$. And

$$l_2 = -\frac{1}{2}[log|G| + \tilde{u}'G^{-1}\tilde{u}] \tag{3.17}$$

3.2.1. E-step of REML EM algorithm

The E-step of the REML EM algorithm requires the conditional distributions $\tilde{u}|\tilde{y}_2$, use the property of conditional distribution of normal distribution, we have

$$\begin{aligned} E\left(\tilde{u}|\tilde{y}_2\right) &= \left(GZ'L_2\right)\left(L_2'HL_2\right)^{-1}\tilde{y}_2 \\ &= GZ'L_2\left(L_2'HL_2\right)^{-1}L_2'\tilde{y} \\ &= GZ'P\tilde{y} \end{aligned} \tag{3.18}$$

$$
\begin{aligned}
var\left(\tilde{u}|\tilde{y}_2\right) &= G - GZ'L_2\left(L_2'HL_2\right)^{-1}L_2'ZG \\
&= G - GZ'PZG
\end{aligned}
\tag{3.19}
$$

where $P = L_2\left(L_2'HL_2\right)^{-1}L_2' = H^{-1} - H^{-1}X(X'H^{-1}X)^{-1}X'H^{-1}$. The E-step involves expectation of l over $\tilde{u}$ given $\tilde{y}_2$ and the k^{th} iteration of $\tilde{\xi}, \tilde{\xi}^{(k)}$. $E\left(l|\tilde{y}_2;\tilde{\xi}^{(k)}\right) = E\left(l_1|\tilde{y}_2;\tilde{\xi}^{(k)}\right) + E\left(l_2|\tilde{y}_2;\tilde{\xi}^{(k)}\right)$ and

$$
\begin{aligned}
E\left(l_1|\tilde{y}_2;\tilde{\xi}^{(k)}\right) =& -\frac{1}{2}\left\{log|\sigma^2 L_2'\Sigma L_2| + E[\left(\tilde{y}'S\tilde{y} - \tilde{u}'Z'S\tilde{y} - \tilde{y}'SZ\tilde{u} + \tilde{u}'Z'SZ\tilde{u}\right)|\tilde{y}_2;\tilde{\xi}^{(k)}]\right\} \\
=& -\frac{1}{2}\left[log|\sigma^2 L_2'\Sigma L_2| + \tilde{y}'S\tilde{y} - \left(\tilde{\mu}^{(k)}\right)'Z'S\tilde{y} - \tilde{y}'SZ\left(\tilde{\mu}^{(k)}\right)\right. \\
&\left. + \left(\tilde{\mu}^{(k)}\right)'Z'SZ\left(\tilde{\mu}^{(k)}\right) + \mathrm{tr}\left(Z'SZV^{(k)}\left(\tilde{u}|\tilde{y}_2\right)\right)\right] \\
=& -\frac{1}{2}[log|\sigma^2 L_2'\Sigma L_2| + \left(\tilde{y} - Z\tilde{\mu}^k\right)'S\left(\tilde{y} - Z\tilde{\mu}^k\right) + \mathrm{tr}\left(Z'SZV^{(k)}\right)]
\end{aligned}
\tag{3.20}
$$

$$
E\left(l_2|\tilde{y}_2;\tilde{\xi}^{(k)}\right) = -\frac{1}{2}[log|G| + \left(\tilde{\mu}^{(k)}\right)'G^{-1}\tilde{\mu}^{(k)} + \mathrm{tr}\left(G^{-1}V^{(k)}\right)] \tag{3.21}
$$

where $\tilde{\mu}^{(k)} = E\left(\tilde{u}|\tilde{y}_2;\tilde{\xi}^{(k)}\right) = G^{(k)}Z'P^{(k)}\tilde{y}$ and $V^{(k)} = V\left(\tilde{u}|\tilde{y}_2;\tilde{\xi}^{(k)}\right) = G^{(k)} - G^{(k)}Z'P^{(k)}ZG^{(k)}$ are conditional expectation and variance of $\tilde{u}$ given $\tilde{y}_2$ at the k^{th} iteration.

3.2.2. M-step for σ^2 and $\tilde{\phi}$

The estimations of σ^2 and $\tilde{\phi}$ are obtained by maximizing of equation 3.20. Noting that , $R = \sigma^2\Sigma(\tilde{\phi})$ and the length of $\tilde{\phi}$ depends on the structure of the residual variance covariance matrix, for example, $\tilde{\phi} = \{\rho\}$ with length 1 for first-order autoregressive AR(1) covariance structure. Derivatives of equation 3.20 with respect to σ^2 and $\tilde{\phi}$ are

$$
\begin{aligned}
\frac{\partial E\left(l_1|\tilde{y}_2;\tilde{\xi}^{(k)}\right)}{\partial\sigma^2} =& -\frac{1}{2}\left\{\frac{n-p}{\sigma^2} - \frac{1}{\left(\sigma^2\right)^2}\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right)'U\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right)\right. \\
&\left. - \frac{1}{\left(\sigma^2\right)^2}\mathrm{tr}\left(Z'UZV^{(k)}\right)\right\}
\end{aligned}
\tag{3.22}
$$

$$
\begin{aligned}
\frac{\partial E\left(l_1|\tilde{y}_2;\tilde{\xi}^{(k)}\right)}{\partial\phi_j} =& -\frac{1}{2}\left\{\mathrm{tr}\left(U\frac{\partial\Sigma}{\partial\phi_j}\right) - \frac{1}{\sigma^2}\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right)'U\frac{\partial\Sigma}{\partial\phi_j}U\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right)\right. \\
&\left. - \frac{1}{\sigma^2}\mathrm{tr}\left(Z'U\frac{\partial\Sigma}{\partial\phi_j}UZV^{(k)}\right)\right\}
\end{aligned}
\tag{3.23}
$$

where $U = L_2(L_2'\Sigma L_2)^{-1}L_2' = \Sigma^{-1} - \Sigma^{-1}X\left(X'\Sigma^{-1}X\right)^{-1}X'\Sigma^{-1}$. Set equation 3.22 to zero we get

$$
(\sigma^2)^{(k+1)} = \frac{1}{n-p}\left\{\left(\tilde{y} - Z\mu^{(k)}\right)'U\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right) + \mathrm{tr}\left(Z'UZV^{(k)}\right)\right\} \tag{3.24}
$$

Equations 3.22 and 3.23 indicate that the updating formulas for σ^2 and $\tilde{\phi}$ depend on each other. In this situation, Meng and Rubin [14] suggested that the Expectation Conditional Maximization (ECM) algorithm can be used to update σ^2 and $\tilde{\phi}$ sequentially, i.e, in the current M-step, σ^2 is estimated with estimation of $\tilde{\phi}$ from the last EM iteration, then $\tilde{\phi}$ could be estimated with current estimation of σ^2. However, the updating formula for $\tilde{\phi}$ that doesn't have closed form solution makes the problem worse. Another iteration solution with each of EM iteration would make the algorithm inefficient. Instead, one cycle of projected gradient descent algorithm is employed to update $\tilde{\phi}$. Then the new update formulates are given by

$$(\sigma^2)^{(k+1)} = \frac{1}{n-p}\left\{\left(\tilde{y} - Z\mu^{(k)}\right)' U^{(k)}\left(\tilde{y} - Z\tilde{\mu}^{(k)}\right) + \mathrm{tr}\left(Z'U^{(k)}ZV^{(k)}\right)\right\} \tag{3.25}$$

where $U^{(k)} = {\Sigma^{(k)}}^{-1} - {\Sigma^{(k)}}^{-1}X\left(X'{\Sigma^{(k)}}^{-1}X\right)^{-1}X'{\Sigma^{(k)}}^{-1}$. And

$$\tilde{d}^{(k+1)} = \tilde{\phi}^{(k)} - \alpha_k\left(\frac{\partial E\left(l_1|\tilde{y}_2;\tilde{\xi}^{(k)}\right)}{\partial\tilde{\phi}}|_{\tilde{\theta}=\tilde{\theta}^{(k)},\sigma^2=(\sigma^2)^{(k+1)}}\right) \tag{3.26}$$

$$\tilde{\phi}^{(k+1)} = argmin_{\tilde{\phi}\in\Phi}\frac{1}{2}||\tilde{\phi} - \tilde{d}^{(k)}||^2 \tag{3.27}$$

where α_k is the step size for gradient descent algorithm at iteration k and Φ is the parameter space of $\tilde{\phi}$. In the case of Σ takes AR(1) covariance structure with $|\rho| < 1$, the updating formula for ρ is

$$\rho^{k+1} = sign(d^{k+1})min(1, abs(d^{k+1})). \tag{3.28}$$

3.2.3. M-step for $\tilde{\gamma}$

Estimations of $\tilde{\gamma}$ are obtained by maximizing of equation 3.21. Noting that $G = G(\gamma)$ is a matrix of γ_{gh} and $\gamma_{gh} = \gamma_{hg}$ for all $h \neq k$ and g,h=1,2,....,b. Derivative of equation 3.21 with respect to $\tilde{\gamma}$ is

$$\begin{aligned}\frac{\partial E\left(l_2|\tilde{y}_2;\theta^{(k)}\right)}{\partial\gamma_{gh}} = &-\frac{1}{2}\left\{\mathrm{tr}\left(G^{-1}\frac{\partial G}{\partial\gamma_{gh}}\right) - \tilde{\mu}^{(k)'}G^{-1}\frac{\partial G}{\partial\gamma_{gh}}G^{-1}\tilde{\mu}^{(k)}\right.\\ &\left. - \mathrm{tr}\left(G^{-1}\frac{\partial G}{\partial\gamma_{gh}}G^{-1}V^{(k)}\right)\right\}\end{aligned} \tag{3.29}$$

The updating formula for γ_{gh} depends on the form of variance matrix G.

Case 1 For linear mixed model contains only one random effect and G has simple form $G = \sigma^2_{\tilde{u}}I_b$, updating equation for $\sigma^2_{\tilde{u}}$ is

$$(\sigma^2_u)^{(k+1)} = \frac{1}{b}[\tilde{\mu}^{(k)'}\tilde{\mu}^{(k)} + \mathrm{tr}(V^{(k)})] \tag{3.30}$$

Case 2 For linear mixed model contains correlated two correlated random effect $\tilde{u}_1, \tilde{u}_2$ that are $b \times 1$ vectors and $cov(u_1, u_2) = \begin{pmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{12} & \gamma_{22} \end{pmatrix}$, the updating formula for γ_{gh} is

$$\gamma_{gh} = \frac{1}{m}(\tilde{\mu}_g^{(k)'}\tilde{\mu}_h^{(k)} + \text{tr}(V_{gh}^k)) \tag{3.31}$$

where $g, h = 1, 2$. See detail derivation in the Appendix B

3.3. Variable Selection Via LASSO

As mentioned earlier, LASSO can reduce the dimension of fixed effect coefficients $\tilde{\beta}$. Consider log likelihood function of $\tilde{y}$ excluding constant term

$$l(\tilde{\beta}; \tilde{y}) = -\frac{1}{2}[log|\hat{H}^{-1}| + (\tilde{y} - X\tilde{\beta})'\hat{H}^{-1}(\tilde{y} - X\tilde{\beta})] \tag{3.32}$$

where $\hat{H}$ is the estimated variance-covariance matrix of y that is obtained from section 3.2. The LASSO method imposes an $l1$ norm on above likelihood function to achieve the goal of variable selection, i.e, instead of minimize $-l(\tilde{\beta}; \tilde{y})$, we minimize

$$L(\beta) = \frac{1}{2}[log|\hat{H}^{-1}| + (\tilde{y} - X\tilde{\beta})'\hat{H}^{-1}(\tilde{y} - X\tilde{\beta})] + \lambda||\tilde{\beta}||_1 \tag{3.33}$$

where $\lambda \in [0, +\infty)$ is the Lagrange multiplier and $||\tilde{\beta}||_1 = \sum_{j=1}^{j=p-1} |\beta_j|$. Notice that when λ is large enough, the $l1$ norm penalty will force $||\tilde{\beta}||_1 = 0$ and when $\lambda = 0$, estimation of $\tilde{\beta}$ is exactly same as the ordinary least square estimation. Hence, we are looking for an appropriate λ value that yields a parsimonious model, i.e., a meaningful model with less fixed effects.

Removing the constant term, equation 3.33 can be rewritten as

$$L(\beta) = \frac{1}{2}||\tilde{y}^* - X^*\tilde{\beta}||^2 + \lambda||\tilde{\beta}||_1 \tag{3.34}$$

where $\tilde{y}^* = D'\tilde{y}$ and and $X^* = D'X$, D is the Cholesky decomposition of $\hat{H}^{-1}$, i.e. $\hat{H}^{-1} = DD'$ and D is lower triangular matrix. Let $X^* = (\tilde{X}_0^*, \tilde{X}_1^*, ..., \tilde{X}_{p-1}^*)$, then derivatives of $L(\beta)$ with respect to β_i with $\beta_j, j \neq i$ fixed are:

$$\frac{\partial L(\tilde{\beta})}{\partial \beta_0} = (\tilde{X}_0^*)'\tilde{X}_0^*\beta_0 - (\tilde{X}_0^*)'(\tilde{y}^* - \tilde{X}_{-0}^*\tilde{\beta}_{-0}) \quad \text{as we don't penalize } \beta_0 \tag{3.35}$$

$$\frac{\partial L(\tilde{\beta})}{\partial \beta_i} = (\tilde{X}_i^*)'\tilde{X}_i^*\beta_i - (\tilde{X}_i^*)'(\tilde{y}^* - \tilde{X}_{-i}^*\tilde{\beta}_{-i}) + \lambda s_i \quad i = 1, 2, .., p-1 \tag{3.36}$$

where X^*_{-i} is the matrix X^* without ith column and $\tilde{\beta}_{-i}$ is $\tilde{\beta}$ without β_i, $s_i \in \partial|\beta_i|, i = 0, 1, 2, .., p-1$. Setting equations 3.35 and 3.36 to zero, β's can be updated coordinate wisely:

$$\beta_0^{(k+1)} = \frac{(\tilde{X}_0^*)'(\tilde{y}^* - \tilde{X}^*_{-0}\tilde{\beta}^{(k+1)}_{-0})}{(\tilde{X}_0^*)'\tilde{X}_0^*} \tag{3.37}$$

$$\beta_i^{(k+1)} = S_{\lambda/||X_i^*||^2}\left(\frac{(\tilde{X}_i^*)'(\tilde{y}^* - \tilde{X}^*_{-i}\tilde{\beta}^{(k+\frac{1}{2})}_{-i})}{(\tilde{X}_i^*)'\tilde{X}_i^*}\right) \quad i = 1, 2, .., p-1 \tag{3.38}$$

where $\tilde{\beta}^{(k+\frac{1}{2})}_{-i} = (\beta_0^{(k+1)}, \beta_1^{(k+1)}, .., \beta_{i-1}^{(k+1)}, \beta_{i+1}^{(k)}, ...\beta_{p-1}^{(k)})$ and $S_\lambda(\beta)$ is soft-threshold operator and

$$S_\lambda(\beta) = \begin{cases} \beta - \lambda & \text{if } \beta > \lambda \\ 0 & \text{if } |\beta| \leq \lambda \\ \beta + \lambda & \text{if } \beta < -\lambda \end{cases}$$

Pathwise Coordinate Descent (PCD) algorithm is used to find best tuning parameter λ. PCD begins with a large λ so that $\tilde{\beta}_{-0} = \tilde{0}$, then the value of λ is reduced a little bit and the coordinate descent is performed until convergence. λ is reduced again, we then run coordinate descent until convergence with the previous solution as initial value. By doing this over and over again, we run coordinate descent over a grid of λ values. The critical question here is which λ is the best. AIC and BIC are the common criteria used to identify the best λ, Zou et al. [21] suggested that BIC is preferred when the the goal is to detect the sparsity of a model. Following their proposed BIC for LASSO definition, the BIC in our case is

$$BIC(X^*\hat{\beta}_\lambda) = \frac{||y^* - X^*\hat{\beta}_\lambda||}{N} + \frac{log(N)}{N}\hat{df}(\lambda) \tag{3.39}$$

where N is total number of observations, $\hat{\beta}_\lambda$ is the estimation of $\tilde{\beta}$ at a given value of λ, and $\hat{df}(\lambda))$ is defined as the degrees of freedom of the LASSO. Zou.etc.[21] also showed that number of nonzero coefficients in the lasso at a given value of λ is an unbiased estimate of the degrees of freedom. The λ that yields smallest BIC value is the best λ.

4. NUMERICAL EXPERIMENT

We have presented the details of the proposed two-stage model in Chapter 3; in this chapter, we present the performance of the proposed method with simulation studies and analysis of a life data set.

4.1. Design of The Simulation

Simulation studies for longitudinal data are designed based on the linear mixed model in the following form

$$\tilde{y}_i = X_i\tilde{\beta} + Z_i\tilde{u} + \tilde{e}_i \tag{4.1}$$

where $i = 1, 2, ..., m$, m is the number of subjects, the number of measurements collected for the i^{th} subject is tp_i. To investigate how the sample size influences the performance of the proposed method, m takes values of 50, 80, 100, and 200. Because dropout is a common concern in longitudinal studies, to make the simulation more close to reality, we consider the following two scenarios in terms of the number of time points: 1. Set the maximum number of time points as 10 and the minimum number of time points as 6 and 2. Set set the maximum number of time points as 10 and the minimum number of time points as 8. By doing so, we can evaluate the effect of the dropout rate on the performance of the proposed method.

Let $\tilde{\beta} = (3, 3.6, 8, -5.4, 5, 2, 0, 0, ..., 0)'$ which is a $p \times 1$ vector and contains 6 non-zero values and $p - 6$ zero values, here p takes values of 20, 40 and 100. The design matrix of fixed effects is $X_i = (X_{i1}, X_{i2}, ..., X_{ip})$ where $X_{i1} = \tilde{1}$ is the design matrix corresponding to intercept β_0, $X_{i2} = (T_i1, T_i2, ...T_{tp_i})'$ is design matrix corresponding to time effect. To make the example closer to reality, we allow X_{i3}, X_{i4} and X_{i5} correlated in the following way: $X_{i3} \sim unif(-0.3, 0.3)$, $X_{i4} \sim N(X_{i3}, 0.5^2)$, $X_{i5} \sim unif(X_{i4} - 1, X_{i4} + 1)$, $X_{i6} \sim N(0, 2^2)$ and $(X_{i7}, X_{i8}, X_{i9})' \sim N(\tilde{0}, \Sigma_0)$, $(X_{i10}, ..., X_{ip}) \sim N(\tilde{0}, I_{p-9})$, where

$$\Sigma_0 = \begin{pmatrix} 0.5 & 0.1 & 0.1 \\ 0.1 & 0.4 & 0.1 \\ 0.1 & 0.1 & 0.6 \end{pmatrix}.$$

The random errors $\tilde{e}_i \sim N(0, \sigma_\epsilon^2 \Sigma_i)$ where $\sigma_\epsilon^2 = 2$ and $(\Sigma_i)_{ts} = (\rho^{|T_{it} - T_{is}|})$ with $\rho = 0.8$ and $t, s = 1, 2, ..., tp_i$. For random effect, two scenarios are considered: (1) model with the random slope only, (2) model with correlated random intercept and random slope.

For the model has random slope only, $\tilde{u} = (u_1, u_2, ..., u_m)' \sim N(\tilde{0}, \sigma_1^2 I_m)$ with $\sigma_u^2 = 0.64$ and corresponding design matrix $Z = bdiag(\tilde{Z}_i), i = 1, 2, 3, ..., m$ and $\tilde{Z}_i = (T_1, T_2, ...T_{tp_i})'$. For the model with the random intercept and random slope, the random intercept $\tilde{u}_1 = (u_{11}, u_{12}, ..., u_{1m})'$ and random slope $\tilde{u}_2 = (u_{21}, u_{22}, ..., u_{2m})'$ follow multivariate normal distribution

$$\begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \tilde{0} \\ \tilde{0} \end{bmatrix}, \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{12} & \gamma_{22} \end{bmatrix} \otimes I_m \right).$$

where $\gamma_{11} = 1.44$, $\gamma_{12} = 0.64$ and $\gamma_{22} = 1$. The design matrix $Z = (Z_1, Z_2)$ where $Z_1 = bdiag(\tilde{1}_{tp_i}), i = 1, 2, ..., m$, $Z_2 = bdiag(\tilde{Z}_i), i = 1, 2, 3, ..., m$ and $\tilde{Z}_i = (T_1, T_2, ...T_{tp_i})'$.

Totally, there are $4 \times 2 \times 3 = 48$ scenarios in the design of simulation, 100 sets of data are generated for each of those scenarios. The update equations for all the parameters and variable selection procedure are given in the Chapter 3.

4.2. Simulation Results

The simulation results contain two parts that correspond to the two steps in the proposed method. The first part contains the results for variance-covariance parameters estimation. In this part, we compare the results yielded by our method (denoted as pEM in the tables) to those obtained from the NR method. We provide the sample mean and sample standard deviation (Std) of estimated variance-covariance parameter vector $\hat{\xi}$ over the 100 data sets for both methods. The sample means are presented in the first row of each cell in Table 4.1-4.3 and Table 4.5-4.7 and Stds are given in the parenthesis under the means. The second part of the simulation results corresponds to the variable selection stage. In this part, average true nonzero rate, average false nonzero rate, average true zero rate, average false nonzero rate, and F-score are used to assess the performance of the second step of our method. F-score is one of the common metrics used to measure the performance of a classification algorithm and is calculated via the following formula:

$$F - score = 2 * \frac{precision * recall}{precision + recall} \tag{4.2}$$

where

$$precision = \frac{\#\ of\ True\ Zero}{\#\ of\ True\ Zero + \#\ of\ False\ Zero} \tag{4.3}$$

$$recall = \frac{\#\ of\ True\ Zero}{\#\ of\ True\ Zero + \#\ of\ False\ Nonzero} \tag{4.4}$$

The higher an F-score, the higher capability of identifying redundant explanatory variables a model has. The variable selection results are presented in Table 4.4 and Table 4.8.

The results presented in Tables 4.1-4.4 are for the model with one random effect. Tables 4.1-4.3 present the results for the variance-covarance parameters estimation corresponding to number of fixed effect p=20,40,and 100 respectively. Table 4.4 contains the variable selection results in all scenarios. When we look closely at each table, we find that, in this simple case, the estimation of variance-covariance parameters values yielded by both methods are always very close to the true values. This is true even when the sample size is small and the drop out rate increases. When we compare the results across Tables 4.1- 4.3, it is easy to see that the results are similar when p (the number of the coefficients) varies from 20 to 100. This further verifies the fact that the REML variance-covariance estimators are invariant to the fixed effect. Table 4.4 shows that, in all scenarios, LASSO successfully identifies all the relevant fixed effects (average true nonzero rate is 100%). On average when p=100, LASSO identifies at least 90 out of 94 irrelevant fixed effects; on average when p=40, at least 31 out of 34 zero coefficients are shrunken to zero; and on average when p=20, more than 12 out of 14 zero coefficients are identified by LASSO.The F-score also shows that LASSO can identify the redundant explanatory variables with very high accuracy in all scenarios. Overall, our method performs well in this simple case.

Table 4.1. Comparison of variance estimation for the model with random slope only with number of coefficients p=20

tp	# of subject	Method	σ_u^2 (0.64)	σ_ϵ^2 (2)	ρ (0.8)
tp=8 $\sim$ 10	m=200	pEM	0.6395	1.9981	0.7995
		(Std)	(0.0616)	(0.1906)	(0.0201)
		NR	0.6394	2.0075	0.8004
		(Std)	(0.0616)	(0.1916)	(0.0201)
	m=100	pEM	0.6413	1.9926	0.7962
		(Std)	(0.0911)	(0.2644)	(0.0255)
		NR	0.6409	2.0131	0.7982
		(Std)	(0.0910)	(0.2678)	(0.0254)
	m=80	pEM	0.6412	1.9860	0.7902
		(Std)	(0.0995)	(0.2105)	(0.0426)
		NR	0.6409	2.0052	0.7951
		(Std)	(0.0990)	(0.2059)	(0.0393)
	m=50	pEM	0.6576	1.9387	0.7730
		(Std)	(0.1187)	(0.2416)	(0.0636)
		NR	0.6566	1.9995	0.7909
		(Std)	(0.1186)	(0.2416)	(0.0512)
tp=6$\sim$ 10	m=200	pEM	0.6378	1.9850	0.7915
		(Std)	(0.0695)	(0.1352)	(0.0296)
		NR	0.6376	1.9955	0.7943
		(Std)	(0.0695)	(0.1290)	(0.0273)
	m=100	pEM	0.6346	2.0112	0.7985
		(Std)	(0.0855)	(0.1844)	(0.0398)
		NR	0.6344	2.0218	0.8017
		(Std)	(0.0854)	(0.1618)	(0.0357)
	m=80	pEM	0.6393	2.0003	0.7928
		(Std)	(0.1168)	(0.2214)	(0.0512)
		NR	0.6388	2.0269	0.8009
		(Std)	(0.1169)	(0.2059)	(0.0426)
	m=50	pEM	0.6488	1.9341	0.7758
		(Std)	(0.1293)	(0.2774)	(0.0667)
		NR	0.6474	2.0188	0.8012
		(Std)	(0.1294)	(0.2746)	(0.0545)

Table 4.2. Comparison of variance estimation for the model with random slope only with number of coefficients p=40

tp	# of subjects	Method	σ_u^2 (0.64)	σ_ϵ^2 (2)	ρ (0.8)
tp=8 $\sim$ 10	m=200	pEM	0.6467	1.9871	0.7956
		(Std)	(0.0696)	(0.1245)	(0.0273)
		NR	0.6465	1.9951	0.7975
		(Std)	(0.0696)	(0.1260)	(0.0257)
	m=100	pEM	0.6300	2.0155	0.7978
		(Std)	(0.0963)	(0.3119)	(0.0318)
		NR	0.6296	2.0382	0.8000
		(Std)	(0.0964)	(0.3135)	(0.0313)
	m=80	pEM	0.6390	1.9807	0.7939
		(Std)	(0.1028)	(0.2727)	(0.0307)
		NR	0.6386	2.0035	0.7961
		(Std)	(0.1028)	(0.2788)	(0.0308)
	m=50	pEM	0.6464	1.9079	0.7826
		(Std)	(0.1211)	(0.3566)	(0.0481)
		NR	0.6458	1.9542	0.7887
		(Std)	(0.1211)	(0.35758)	(0.0434)
tp=6$\sim$ 10	m=200	pEM	0.6341	1.9894	0.7973
		(Std)	(0.0748)	(0.1336)	(0.0275)
		NR	0.6340	1.9942	0.7986
		(Std)	(0.0747)	(0.1292)	(0.0243)
	m=100	pEM	0.6319	1.9785	0.7947
		(Std)	(0.0986)	(0.2521)	(0.0266)
		NR	0.6316	1.9991	0.7968
		(Std)	(0.0986)	(0.2559)	(0.0261)
	m=80	pEM	0.6383	1.9827	0.7953
		(Std)	(0.1081)	(0.2640)	(0.0272)
		NR	0.6379	2.0076	0.7978
		(Std)	(0.1081)	(0.3708)	(0.0269)
	m=50	pEM	0.6440	1.9583	0.7900
		(Std)	(0.1494)	(0.3708)	(0.0417)
		NR	0.6433	2.0074	0.7958
		(Std)	(0.1496)	(0.3769)	(0.0388)

Table 4.3. Comparison of variance estimation for the model with random slope only with number of coefficient p=100

Time points	# of subject	Method	σ_u^2 (0.64)	σ_e^2 (2)	ρ (0.8)
tp=8 ∼10	m=200	pEM	0.6397	1.9892	0.7969
		(Std)	(0.0667)	(0.1724)	(0.0181)
		NR	0.6396	1.9975	0.7978
		(Std)	(0.0667)	(0.1707)	(0.0177)
	m=100	pEM	0.6564	1.9752	0.7908
		(Std)	(0.0846)	(0.1912)	(0.0443)
		NR	0.6560	1.9992	0.7979
		(Std)	(0.0847)	(0.1768)	(0.0375)
	m=80	pEM	0.6513	1.9728	0.7859
		(Std)	(0.0942)	(0.2143)	(0.0587)
		NR	0.6509	1.9992	0.7943
		(Std)	(0.0941)	(0.2123)	(0.0483)
	m=50	pEM	0.6285	1.9128	0.7607
		(Std)	(0.1302)	(0.2388)	(0.0653)
		NR	0.6269	1.9987	0.7856
		(Std)	(0.1300)	(0.2610)	(0.0643)
tp=6∼ 10	m=200	pEM	0.6373	1.9944	0.7956
		(Std)	(0.0682)	(0.1556)	(0.0360)
		NR	0.6371	2.0039	0.7989
		(Std)	(0.0683)	(0.1435)	(0.0282)
	m=100	pEM	0.6529	1.9637	0.7897
		(Std)	(0.1037)	(0.2119)	(0.0508)
		NR	0.6525	1.9875	0.7973
		(Std)	(0.1037)	(0.2011)	(0.0428)
	m=80	pEM	0.6420	1.9756	0.7822
		(Std)	(0.1018)	(0.2245)	(0.0587)
		NR	0.6408	2.0410	0.8020
		(Std)	(0.1021)	(0.2134)	(0.0509)
	m=50	pEM	0.6475	1.9499	0.7752
		(Std)	(0.1165)	(0.3025)	(0.0756)
		NR	0.6453	2.0707	0.8102
		(Std)	(0.1166)	(0.3341)	(0.0695)

Table 4.4. Variable select results for the model with random slope only

		# of subjects	True Nonzero Rate	False Nonzero Rate	True Zero Rate	False Zero Rate	F score
tp=8 ∼10	p=100	m=200	100% (6/6)	2.04% (1.92/94)	97.96% (92.08/94)	0% (0/6)	0.9897
		m=100	100% (6/6)	3.51% (3.30/94)	96.49% (90.70/94)	0% (0/6)	0.9821
		m=80	100% (6/6)	4.02% (3.78/94)	95.98% (90.22/94)	0% (0/6)	0.9795
		m=50	100% (6/6)	4.15% (3.90/94)	95.85% (90.10/94)	0% (0/6)	0.9788
	p=40	m=200	100% (6/6)	4.88% (1.66/34)	95.12% (32.34/34)	0% (0/6)	0.9750
		m=100	100% (6/6)	5.41% (1.84/34)	94.59% (32.16/34)	0% (0/6)	0.9722
		m=80	100% (6/6)	5.65% (1.92/34)	94.35% (32.08/34)	0% (0/6)	0.9709
		m=50	100% (6/6)	6.62% (2.25/34)	93.38% (31.75/34)	0% (0/6)	0.9658
	p=20	m=200	100% (6/6)	11.14% (1.56/14)	88.86% (12.44/14)	0% (0/6)	0.9410
		m=100	100% (6/6)	11.29% (1.58/14)	88.71% (12.42/14)	0% (0/6)	0.9402
		m=80	100% (6/6)	5.43% (0.76/14)	94.57% (13.24/14)	0% (0/6)	0.9721
		m=50	100% (6/6)	8.57% (1.20/14)	91.43% (12.80/14)	0% (0/6)	0.9552
tp=6 ∼ 10	p=100	m=200	100% (6/6)	2.23% (2.10/94)	97.77% (91.90/94)	0% (0/6)	0.9887
		m=100	100% (6/6)	3.60% (3.38/94)	96.40% (90.62/94)	0% (0/6)	0.9817
		m=80	100% (6/6)	3.60% (3.38/94)	96.40% (90.62/94)	0% (0/6)	0.9817
		m=50	100% (6/6)	4.06% (3.82/94)	95.94% (90.18/94)	0% (0/6)	0.9793
	p=40	m=200	100% (6/6)	4.68% (1.59/34)	95.32% (32.41/34)	0% (0/6)	0.9760
		m=100	100% (6/6)	5.68% (1.93/34)	94.32% (32.07/34)	0% (0/6)	0.9708
		m=80	100% (6/6)	6.15% (2.09/34)	93.85% (31.91/34)	0% (0/6)	0.9683
		m=50	100% (6/6)	7.43% (2.53/34)	92.57% (31.47/34)	0% (0/6)	0.9614
	p=20	m=200	100% (6/6)	5.93% (0.83/14)	94.07% (13.17/14)	0% (0/6)	0.9694
		m=100	100% (6/6)	6.29% (0.88/14)	93.71% (13.12/14)	0% (0/6)	0.9675
		m=80	100% (6/6)	7.64% (1.07/14)	92.36% (12.93/14)	0% (0/6)	0.9603
		m=50	100% (6/6)	9.43% (1.32/14)	90.57% (12.68/14)	0% (0/6)	0.9505

Tables 4.5-4.8 give the full simulation results for the model with correlated random intercept and random slope. Tables 4.5-4.7 present the results for variance-covarance parameters estimation corresponding to number of fixed effect p=20, 40, and 100 respectively. Table 4.8 contains the variable selection results. Table 4.5 shows that when the sample size is large, the estimated variance-covariance parameter values produced by our method are much closer to the true values than the NR method. However, when the sample size is small, equaling 50 in our case, the accuracy of our method is slightly lower than the NR method. The same conclusion can be drawn from Tables 4.6 and 4.7, where number of the coefficients p varies from 20 to 100. Therefore, the fact that the REML variance-covariance estimators are invariant to the fixed effect is also true in this complicated case . Table 4.8 shows that, in all scenarios, LASSO successfully identifies all the relevant fixed effects (true nonzero rate is 100%). On average when p=100, LASSO identifies at least 93 out of 94

irrelevant fixed effects; on average when p=40, at least 29 out of 34 zero coefficients are shrunken to zero; on average when p=20, more than 11 out of 14 zero coefficients are identified by LASSO. The F-score also shows that LASSO can accurately identify the redundant explanatory variables in all scenarios. Comparing the results in this case to the results of the random slope only model, we can tell that the performance of our method is affected by the complexity of the variance-covariance structure.

Table 4.5. Comparison of variance estimation for the model with correlated random intercept and random slope with number of coefficients p=20

tp	# of subjects	Method	γ_{11} (1.44)	γ_{12} (0.64)	γ_{22} (1)	σ^2_ϵ (2)	ρ (0.8)
tp=8 $\sim$ 10	m=200	pEM	1.4773	0.6241	1.0023	1.9773	0.7883
		(Std)	(0.6265)	(0.1015)	(0.0306)	(0.4343)	(0.0454)
		NR	1.7114	0.5013	0.9995	1.8140	0.7720
		(Std)	(0.5245)	(0.1442)	(0.0292)	(0.3632)	(0.0447)
	m=100	pEM	1.4618	0.6367	0.9971	2.0000	0.7845
		(Std)	(0.7131)	(0.1524)	(0.0422)	(0.5650)	(0.0604)
		NR	1.7496	0.4793	1.0081	1.7536	0.7585
		(Std)	(0.6851)	(0.1618)	(0.0423)	(0.4784)	(0.0638)
	m=80	pEM	1.6079	0.6079	1.0095	1.8936	0.7721
		(Std)	(0.7552)	(0.1722)	((0.0493)	(0.5490)	(0.0609)
		NR	1.6975	0.4968	1.0100	1.8247	0.7648
		(Std)	(0.7727))	(0.1793)	(0.0476)	(0.5290)	(0.0650)
	m=50	pEM	1.9483	0.5978	1.0020	1.6607	0.7336
		(Std)	(0.9092)	(0.1921)	(0.0567)	(0.5831)	(0.0816)
		NR	2.0711	0.4333	1.0014	1.5987	0.7244
		(Std)	(0.7763)	(0.1883)	(0.0547)	(0.5258)	(0.0860)
tp=6 $\sim$ 10	m=200	pEM	1.4597	0.6629	0.9900	2.0419	0.7990
		(Std)	(0.5560)	(0.1135)	(0.0302)	(0.4046)	(0.0390)
		NR	1.6408	0.5372	0.9916	1.9031	0.7853
		(Std)	(0.5251)	(0.1441)	(0.0290)	(0.3665)	(0.0406)
	m=100	pEM	1.5061	0.6039	1.0059	1.9386	0.7761
		(Std)	(0.7153)	(0.1548)	(0.0405)	(0.5739)	(0.0651)
		NR	1.7493	0.4674	1.0115	1.7398	0.7519
		(Std)	(0.6544)	(0.1562)	(0.0399)	(0.4974)	(0.0701)
	m=80	pEM	1.6252	0.6133	1.0096	1.8389	0.7660
		(Std)	(0.7612)	(0.1818)	(0.0520)	(0.5055)	(0.0644)
		NR	1.7679	0.4710	1.0137	1.7549	0.7552
		(Std)	(0.7065)	(0.1714)	(0.0497)	(0.4810)	(0.0694)
	m=50	pEM	2.0006	0.5742	1.0082	1.5989	0.7235
		(Std)	(0.9204)	(0.2326)	(0.0676)	(0.6197)	(0.0781)
		NR	1.9261	0.4592	1.0143	1.6898	0.7442
		(Std)	(0.8676)	(0.1982)	(0.0629)	(0.5727)	(0.0712)

Table 4.6. Comparison of variance estimation for the model with correlated random intercept and random slope with number of coefficients p=40

tp	# of subjects	Method	γ_{11} (1.44)	γ_{12} (0.64)	γ_{22} (1)	σ^2_ϵ (2)	ρ (0.8)
tp=8 $\sim$ 10	m=200	pEM	1.3699	0.6475	0.9979	2.0744	0.7989
		(Std)	(0.5689)	(0.1175)	(0.0333)	(0.4205)	(0.0427)
		NR	1.5540	0.5363	1.0000	1.9617	0.7872
		(Std)	(0.5419)	(0.1513)	(0.0336)	(0.4289)	(0.0465)
	m=100	pEM	1.5507	0.6302	0.9988	1.9194	0.7800
		(Std)	(0.7199)	(0.1552)	(0.0443)	(0.5054)	(0.0561)
		NR	1.7697	0.5009	0.9971	1.7692	0.7625
		(Std)	(0.6677)	(0.1753)	(0.0460)	(0.4627)	(0.0587)
	m=80	pEM	1.5139	0.6128	1.0001	1.9694	0.7844
		(Std)	(0.7815)	(0.1717)	(0.0458)	(0.5711)	(0.0539)
		NR	1.6782	0.4961	1.0025	1.8261	0.7732
		(Std)	(0.7434	(0.1874)	(0.0457	(0.4283)	(0.0483)
	m=50	pEM	1.9847	0.5991	1.0018	1.6304	0.7309
		(Std)	(0.9379)	(0.2265)	(0.0661)	(0.5613)	(0.0747)
		NR	2.0366	0.4411	1.0043	1.6864	0.74044
		(Std)	(0.8223)	(0.1882)	(0.0529)	(0.5960)	(0.0756)
tp=6 $\sim$ 10	m=200	pEM	1.4597	0.6371	1.0000	2.0273	0.7920
		(Std)	(0.5819)	(0.1004)	(0.0289)	(0.4652)	(0.0441)
		NR	1.6884	0.4936	1.0050	1.8435	0.7733
		(Std)	(0.5404)	(0.1158)	(0.0273)	(0.3844)	(0.0427)
	m=100	pEM	1.4679	0.6314	1.0052	1.9834	0.7839
		(Std)	(0.7536)	(0.1429)	(0.0437)	(0.5404)	(0.0574)
		NR	1.7386	0.5048	1.0070	1.8034	0.7650
		(Std)	(0.6972)	(0.1758)	(0.0452)	(0.4403)	(0.0610)
	m=80	pEM	1.7608	0.6325	0.9949	1.7883	0.7613
		(Std)	(0.9004)	(0.1581)	(0.0459)	(0.4886)	(0.0642)
		NR	1.9761	0.4773	0.9966	1.7464	0.7552
		(Std)	(0.8319)	(0.1645)	(0.0465)	(0.4700)	(0.0707)
	m=50	pEM	2.1079	0.5953	1.0146	1.5342	0.7114
		(Std)	(0.9414)	(0.2380)	(0.0695)	(0.5447)	(0.0782)
		NR	2.0949	0.4476	1.0163	1.6048	0.7272
		(Std)	(0.8892)	(0.2107)	(0.0672)	(0.4943)	(0.0844)

Table 4.7. Comparison of variance estimation for the model with correlated random intercept and random slope with number of coefficients p=100

tp	# of subjects	Method	γ_{11} (1.44)	γ_{12} (0.64)	γ_{22} (1)	σ^2_ϵ (2)	ρ (0.8)
tp=8 $\sim$ 10	m=200	pEM	1.4057	0.6353	1.0030	1.9924	0.7913
		(Std)	(0.5666)	(0.1094)	(0.0268)	(0.3998)	(0.0437)
		NR	1.6220	0.5120	1.0042	1.8583	0.7773
		(Std)	(0.4868)	(0.1318)	(0.0278)	(0.3601)	(0.0435)
	m=100	pEM	1.3595	0.6586	1.0022	2.0606	0.7979
		(Std)	(0.5815)	(0.0982)	(0.0311)	(0.3896)	(0.0391)
		NR	1.5135	0.5660	1.0049	1.9624	0.7891
		(Std)	(0.5577)	(0.1591)	(0.0312)	(0.3597)	(0.0395)
	m=80	pEM	1.6250	0.6098	1.0085	1.8391	0.7653
		(Std)	(0.8041)	(0.1627)	(0.0445)	(0.5495)	(0.0620)
		NR	1.7439	0.4950	1.0068	1.7829	0.7608
		(Std)	(0.6845)	(0.1836)	(0.0457)	(0.4886)	(0.0598)
	m=50	pEM	2.2636	0.5608	1.0225	1.4211	0.6901
		(Std)	(0.9102)	(0.2274)	(0.0669)	(0.4307)	(0.0781)
		NR	2.0974	0.4447	1.0176	1.5973	0.7162
		(Std)	(0.9259)	(0.1981)	(0.0586)	(0.6224)	(0.0963)
tp=6 $\sim$ 10	m=200	pEM	1.3370	0.6476	1.0018	2.1304	0.8033
		(Std)	(0.5921)	(0.1068)	(0.0320)	(0.4780)	(0.0449)
		NR	1.5842	0.5336	1.0044	1.9323	0.7844
		(Std)	(0.5681)	(0.1539)	(0.0310)	(0.4315)	(0.0478)
	m=100	pEM	1.5881	0.6254	1.0021	1.8844	0.7704
		(Std)	(0.7915)	(0.1607)	(0.0494)	(0.5651)	(0.0638)
		NR	1.8939	0.4632	1.0058	1.6658	0.7474
		(Std)	(0.6866)	(0.1494)	(0.0536)	(0.4114)	(0.0599)
	m=80	pEM	1.7318	0.6249	0.9991	1.7782	0.7528
		(Std)	(0.8223)	(0.1637)	(0.0486)	(0.5579)	(0.0703)
		NR	1.8547	0.4806	1.0002	1.7313	0.7492
		(Std)	(0.7365)	(0.1709)	(0.0474)	(0.5000)	(0.0730)
	m=50	pEM	2.2161	0.5539	1.0097	1.3446	0.6742
		(Std)	(0.9364)	(0.2479)	(0.0646)	(0.4906)	(0.0826)
		NR	1.8639	0.4344	1.0103	1.6478	0.7329
		(Std)	(0.9170)	(0.2044)	(0.0645)	(0.5595)	(0.0922)

Table 4.8. Variable selection results for the model with correlated random intercept and random slope

		# of subjects	True Nonzero Rate	False Nonzero Rate	True Zero Rate	False Zero Rate	F score
tp=8 ∼ 10	p=100	m=200	100% (6/6)	4.66% (4.38/94)	95.34% (89.62/94)	0% (0/6)	0.9761
		m=100	100% (6/6)	4.67% (4.39/94)	95.33% (89.61/94)	0% (0/6)	0.9761
		m=80	100% (6/6)	6.63% (6.23/94)	93.37% (87.77/94)	0% (0/6)	0.9657
		m=50	100% (6/6)	9.64% (9.06/94)	90.36% (84.94/94)	0% (0/6)	0.9494
	p=40	m=200	100% (6/6)	9.71% (3.30/34)	90.29% (30.70/34)	0% (0/6)	0.9490
		m=100	100% (6/6)	10.65% (3.63/34)	89.35% (30.37/34)	0% (0/6)	0.9438
		m=80	100% (6/6)	11.06% (3.76/34)	88.94% (30.24/34)	0% (0/6)	0.9415
		m=50	100% (6/6)	12.50% (4.25/34)	87.50% (29.75/34)	0% (0/6)	0.9333
	p=20	m=200	100% (6/6)	8.14% (1.14/14)	91.86% (12.86/14)	0% (0/6)	0.9576
		m=100	100% (6/6)	12.18% (1.71/14)	87.82% (12.29/14)	0% (0/6)	0.9352
		m=80	100% (6/6)	12.74% (1.78/14)	86.86% (12.16/14)	0% (0/6)	0.9297
		m=50	100% (6/6)	14.07% (1.97/14)	85.93% (12.03/14)	0% (0/6)	0.9243
tp=6 ∼ 10	p=100	m=200	100% (6/6)	5.18% (4.87/94)	94.82% (89.13/94)	0% (0/6)	0.9734
		m=100	100% (6/6)	6.33% (5.95/94)	93.67% (88.05/94)	0% (0/6)	0.9673
		m=80	100% (6/6)	7.50% (7.05/94)	92.50% (86.95/94)	0% (0/6)	0.9610
		m=50	100% (6/6)	11.37% (10.68/94)	88.63% (83.32/94)	0% (0/6)	0.9397
	p=40	m=200	100% (6/6)	10.32% (3.51/34)	89.68% (30.49/34)	0% (0/6)	0.9456
		m=100	100% (6/6)	11.41% (3.88/34)	88.59% (30.12/34)	0% (0/6)	0.9395
		m=80	100% (6/6)	11.53% (3.92/34)	88.47% (30.08/34)	0% (0/6)	0.9388
		m=50	100% (6/6)	12.79% (4.35/34)	87.21% (29.65/34)	0% (0/6)	0.9317
	p=20	m=200	100% (6/6)	13.79% (1.93/14)	86.21% (12.07/14)	0% (0/6)	0.9259
		m=100	100% (6/6)	14.57% (2.04/14)	85.43% (11.96/14)	0% (0/6)	0.9214
		m=80	100% (6/6)	15.00% (2.10/14)	85.00% (11.90/14)	0% (0/6)	0.9189
		m=50	100% (6/6)	16.57% (2.32/14)	83.43% (11.68/14)	0% (0/6)	0.9097

4.3. Life Data Analysis

In this section, we apply the proposed method to a life data set that is provided by the researchers from the North Dakota State University Department of Agricultural and Biosystems Engineering.

In agriculture, the level of water and temperature in the soil are believed to have critical impact on the quality of tomato crops. Drip irrigation under mulch is one way to control both soil moisture and temperature. In this type of irrigation system, a soil moisture sensor is used to control the drip system so that it can provide irrigation based on the plants' needs. A two-factor factorial design with 3 replications in an experiment field was employed to investigate the effect of the irrigation system and the mulch type on the quality of the ripe tomato fruit under

Table 4.9. List of variables collected during the field experiment

Notation	
PH	Measurement of ripe tomato PH which is the response variable we are interested in
Week	Week at which tomato was harvested, it takes value 1,2,...,7
Irrigation	Irrigation system, either Irrigation (DI) or No Irrigation (NI))
Mulch	Mulch type. There were three type of mulch: Clear Plastic (CP), Black Plastic (BP), Fabric Mulch (PF), plus No Mulch (NM)
T_{15}	Weekly average soil temperature at 15cm depth underground
T_{30}	Weekly average soil temperature at 30cm depth underground
R_{15}	Weekly average soil resistance at 15cm depth underground
R_{30}	Weekly average soil resistance at 30cm depth underground
P_{15}	Weekly average absolute soil water potential at 15cm depth underground
P_{30}	Weekly average absolute soil water potential at 30cm depth underground

the climate conditions in Fargo, North Dakota. The layout of the design is showed in Figure 4.1. The experiment field was divided into 24 plots; two soil moisture sensors and two soil temperature sensors were installed at 15 cm and 30 cm in each plot to monitor soil moisture and soil temperature variations. Soil resistance and soil temperature readings were collected every 15 minutes throughout the season. Soil water potential was calculated based on the soil resistance readings. Any ripe tomatoes were harvested weekly from each plot during the harvest season, and the PH value of the tomatoes from each plot was measured. The variables collected during this experiment study is listed in Table 4.9.

Figure 4.1. Layout of the two-factor factorial design

An ANCOVA model with random intercept and random slope is employed to detect the factors that are important to PH value of ripe tomato. For the random effect model, instead of setting a unique random intercept/slope coefficient for each subject which would results in too many parameters, we assume random intercept/slope coefficients follows certain distribution, such as normal distribution which only has two parameters involved. The model can be expressed as:

$$
\begin{aligned}
PH_{ij} \quad = \quad & \beta_0 + \beta_1 Week_{ij} + \beta_{21}\mathbb{1}_{\{Much_i=CP\}} + \beta_{22}\mathbb{1}_{\{Much_i=BP\}} + \beta_{23}\mathbb{1}_{\{Much_i=PF\}} \\
& + \beta_3 \mathbb{1}_{\{Irrigation_i=DI\}} + \beta_{41}\mathbb{1}_{\{Much_i=CP\}}\mathbb{1}_{\{Irrigation_i=DI\}} \\
& + \beta_{42}\mathbb{1}_{\{Much_i=BP\}}\mathbb{1}_{\{Irrigation_i=DI\}} + \beta_{43}\mathbb{1}_{\{Much_i=PF\}}\mathbb{1}_{\{Irrigation_i=DI\}} \\
& + \beta_5 T_{15ij} + \beta_6 T_{30ij} + \beta_7 R_{15ij} + \beta_8 R_{30ij} + \beta_9 P_{15ij} + \beta_{10} P_{30ij} + u_{0i} + u_{1i} Week_{ij} + \epsilon_{ij}
\end{aligned}
$$

Where $i = 1, 2, .., 24$, $j = T_1, T_2, ..., T_{tp_i}$ and tp_i is total number of weeks tomato was harvest for the i^{th} plot during throughout the season and tp_i varies from 5 to 7. We assume random intercept u_{0i} follows $N(0, \sigma^2_{u0})$, random slope u_{1i} follows $N(0, \sigma^2_{u1})$, and the error term $\tilde{\epsilon}_i$ follows $N(0, \sigma^2_\epsilon \Sigma_i)$ where $(\Sigma_i)_{st} = \rho^{|T_{is} - T_{it}|}$. We further assume that u_{0i}, u_{1i} and $\tilde{\epsilon}_i$ are mutually independent.

Table 4.10. Variance-covariance parameter estimation for tomato PH value

method	$\hat{\sigma}^2_{u0}$	$\hat{\sigma}^2_{u1}$	$\hat{\sigma}^2_\epsilon$	$\hat{\rho}$
pEM	0.0039	1.95e-07	0.0140	0.3816
NR	0.0047	9.23e-13	0.0128	0.2974

Table 4.11. Variable selection results for tomato PH value

Intercept	Week	MulchCP	MulchBP	MulchPF
4.0514	-0.0024	-	-	-
IrrigationDI	MulchCP*IrrigationDI	MulchBP*IrrigationDI	MulchPF*IrrigationDI	T_{15}
-	-	-	-	-
T_{30}	R_{15}	R_{30}	P_{15}	P_{30}
-	-	-	-	0.0011

Table 4.10 shows that the variance-covariance parameters estimated by our method are similar to those estimated by the NR method, both showing that the random slope is zero, which implies the impact of growth time (week) on the PH value is uniform across all the experiment units. In addition, Table 4.11 shows the LASSO identifies the variables week and soil water potential at 30 cm depth underground as important variables. LASSO selects the explanatory variable P_{30} instead of P_{15} because the root of the tomato plant at 30cm to 40cm depth absorbs more water than other parts of the root. The existence of multicolinearity between P_{30} and other explanatory variables may be another reason that other explanatory variables are not selected. The multicolinearity is evidenced in Tables 4.12 and 4.13. The correlation coefficient matrix in Table 4.12 shows that P_{30} is highly correlated with R_{30} and moderately correlated with R_{15}; consequently, the effect of R_{30} and

R_{15} could be absorbed by P_{30}. To detect the correlation between P_{30} and Mulch, two multinomial linear models with mulch as the response variable are fitted, where the null model has T_{15} and T_{30} as explanatory variables and alternative model contains P_{30} as additional explanatory variable. The deviance test shown in Table 4.13 indicates that P_{30} is significantly correlated with Mulch.

Table 4.12. Pairwise correlation among soil variables

	R_{15}	P_{15}	T_{15}	R_{30}	P_{30}	T_{30}
R_{15}	1.0000	-0.9968	-0.2767	0.5709	-0.5577	-0.2972
P_{15}		1.0000	0.2429	-0.5766	0.5675	0.2660
T_{15}			1.0000	-0.1514	0.0928	0.9608
R_{30}				1.0000	-0.9929	-0.1908
P_{30}					1.0000	0.1320
T_{30}						1.0000

Table 4.13. Deviance test to detect correlation between mulch and P_{30}

Model	Explanatory Variables	Residual deviance	Change in deviance	P-value
Null	T_{15}, T_{30}	420.7092	18.3517	< 0.0001
Alternative	T_{15}, T_{30}, P_{30}	402.3575		

5. DISCUSSION

Like the Restricted Maximum Likelihood (REML) estimator for the linear mixed effect models, the model estimation procedure proposed in this work for LASSO regression of repeated measurements involves two estimation stages: 1. Estimation of the variance-covariance components of the model via EM algorithm and 2. LASSO estimation of deterministic coefficients of model covariates with the estimated variance-covariance components derived in the first stage. For the M-step in the Expectation Maximization (EM) algorithm, we use the gradient descent method instead of the classical NR method, which may avoid computing the variance-covariance matrix of high dimension. Our numerical experiment indicates that, for the first stage, the result of using the gradient descent method is quite comparable to that of NR method and outperforms the NR method when the variance-covariance matrix of the model has a more complex structure. It is noteworthy that in some cases, even when the NR method fails, the gradient descent method is convergence-guaranteed. For the second stage, our numeric experiment shows that the LASSO estimation could successfully identify all the relevant and most of the irrelevant explanatory variables. We also find some limitations of the proposed model estimation procedure. Success of the procedure sometimes relies on a delicate choice of tuning parameters in both the gradient descent method and the LASSO estimation, which largely result from the two estimation methods. The model assumption of Gaussian distribution plays a critical role in both the E-step of EM-algorithm and LASSO estimation. Although, theoretically speaking, it is possible to extend the estimation procedure to non-Gaussian linear models, much more effort is needed in the E-step and the LASSO estimation involved in the proposed estimation procedure.

REFERENCES

[1] EML Beale, MG Kendall, and DW Mann. The discarding of variables in multivariate analysis. *Biometrika*, 54(3-4):357–366, 1967.

[2] Leo Breiman. Heuristics of instability and stabilization in model selection. *Annals of Statistics*, 24(6):2350–2383, 1996.

[3] Robert R Corbeil and Shayle R Searle. Restricted maximum likelihood (reml) estimation of variance components in the mixed model. *Technometrics*, 18(1):31–38, 1976.

[4] Peter J Diggle. An approach to the analysis of repeated measurements. *Biometrics*, pages 959–971, 1988.

[5] M Efroymson. Stepwise regression–a backward and forward look. *Florham Park, New Jersey*, 1966.

[6] Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

[7] Yingying Fan and Runze Li. Variable selection in linear mixed effects models. *Annals of statistics*, 40(4):2043, 2012.

[8] Jean-Louis Foulley, Florence Jaffrézic, and Christèle Robert-Granié. Em-reml estimation of covariance parameters in gaussian mixed models for longitudinal data analysis. *Genetics Selection Evolution*, 32(2):129–141, 2000.

[9] David A Harville. Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American statistical association*, 72(358):320–338, 1977.

[10] David A. Harville. *Matrix algebra from a statistician's perspective*. Springer, New York, 1997.

[11] Lan Lan. *Variable Selection in Linear Mixed Model for Longitudinal Data*. PhD thesis, Department of Statistics, North Carolina State University, 2006.

[12] Kenneth Lange. A gradient algorithm locally equivalent to the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(2):425–437, 1995.

[13] Bingqing Lin, Zhen Pang, and Jiming Jiang. Fixed and random effects selection by reml and pathwise coordinate optimization. *Journal of Computational and Graphical Statistics*, 22(2):341–355, 2013.

[14] Xiao Li Meng and Donald B Rubin. Maximum likelihood estimation via the ecm algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

[15] Xiao Ni, Daowen Zhang, and Hao Helen Zhang. Variable selection for semiparametric mixed models in longitudinal studies. *Biometrics*, 66(1):79–88, 2010.

[16] HD Patterson. Maximum likelihood estimation of components of variance. In *Proceeding Eight International Biometric Conference.* Biometric Soc., 1975.

[17] SN Rai and DE Matthews. Improving the em algorithm. *Biometrics*, pages 587–591, 1993.

[18] David A Sprott. Marginal and conditional sufficiency. *Biometrika*, 62(3):599–605, 1975.

[19] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[20] Arunas Petras Verbyla. A conditional derivation of residual maximum likelihood. *Australian Journal of Statistics*, 32(2):227–230, 1990.

[21] Hui Zou, Trevor Hastie, Robert Tibshirani, et al. On the "degrees of freedom" of the lasso. *The Annals of Statistics*, 35(5):2173–2192, 2007.

APPENDIX A. MATRIX PROPERTIES AND DERIVATIVES

A.1. Conditional Distribution Between Two Vectors

if $(\tilde{x}', \tilde{y}')'$ has joint distribtuion $N_p(\tilde{\mu}, \Sigma)$ where $\mu = (\mu_x', \mu_y')$, and $\Sigma = \begin{pmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy} & \Sigma_y \end{pmatrix}$, and Σ_y is nonsigular, then for a given value of $\tilde{y}$

$$\tilde{x}|\tilde{y} \sim N(\tilde{\mu}_x + \Sigma_{xy}\Sigma_y^{-1}(\tilde{y} - \tilde{\mu}_y), \Sigma_x - \Sigma_{xy}\Sigma_y^{-1}\Sigma_{xy})$$

A.2. Quadratic Form

If $\tilde{y} \sim N(\tilde{\mu}, \Sigma)$ and A is a symmetric matrix, then

$$E(\tilde{y}'A\tilde{y}) = \tilde{\mu}'A\tilde{\mu} + \text{tr}(AV)$$

A.3. Properties of Vec Operator and Kronecker Products

If A and B are square matrices, C is same dimension as A then

$$(vec(A))'(B \otimes C)vec(A) = \text{tr}(B'A'CA)$$

Proof see Harville [10]

A.4. Derivatives with Matrices

Given a nonsingular $p \times p$ matrix $lA = A(x)$ whose elements depend on x and is continuously differentiable over x, then

$$\frac{\partial A^{-1}}{\partial x_j} = -A^{-1}\frac{\partial A}{\partial x_j}A^{-1}$$

$$\frac{\partial \log|A|}{\partial x_j} = \text{tr}\left(A^{-1}\frac{\partial A}{\partial x_j}\right)$$

A.5. Derivatives of $log|L_2'\Sigma L_2|$

$$\begin{aligned} \frac{\partial log|L_2'\Sigma L_2|}{\partial \rho} &= \text{tr}\left\{(L_2'\Sigma L_2)^{-1}L_2'\frac{\partial \Sigma}{\partial \rho}L_2\right\} \\ &= \text{tr}\left\{L_2(L_2'\Sigma L_2)^{-1}L_2'\frac{\partial \Sigma}{\partial \rho}\right\} \\ &= \text{tr}(U\frac{\partial \Sigma}{\partial \rho}) \end{aligned}$$

where $U = L_2(L_2'\Sigma L_2)^{-1}L_2'$.

APPENDIX B. UPDATING FORMULA

Let $G_0 = \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{12} & \gamma_{22} \end{bmatrix}$ and $G_0^{-1} = \begin{bmatrix} \gamma^{11} & \gamma^{12} \\ \gamma^{12} & \gamma^{22} \end{bmatrix}$, then $G = G_0 \otimes I_m$ and $G^{-1} = G_0^{-1} \otimes I_m$.
Define $\mathcal{U}^{(k)}$ such that $\text{vec}(\mathcal{U}^{(k)}) = (\tilde{\mu}_1^{(k)'}, \tilde{\mu}_2^{(k)'})' = \tilde{\mu}^{(k)}$.

The M-step for γ_{gh} is given by

$$\frac{\partial E\left(l_2|\tilde{y}_2;\theta^{(k)}\right)}{\partial \gamma_{gh}} = -\frac{1}{2}\left\{\text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}\right) - \tilde{\mu}^{(k)'}G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}\tilde{\mu}^{(k)} - \text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}V^{(k)}\right)\right\}$$

The first term is given by

$$\begin{aligned} \text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}\right) &= \text{tr}\left((G_0^{-1}\otimes I_m)(\frac{\partial G_0}{\partial \gamma_{gh}}\otimes I_m)\right) \\ &= \text{tr}\left(G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}\otimes I_m\right) \\ &= m\,\text{tr}\left(G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}\right) \end{aligned}$$

When g=h, $\frac{\partial G_0}{\partial \gamma_{gh}}$ is matrix of zero except the (g,g) position, then $\text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}\right) = m(G_0^{-1})_{gg}$.
When g $\neq$ h, $\frac{\partial G_0}{\partial \gamma_{gh}}$ is matrix of zero except the (g,h) and (h,g) positions, then $\text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}\right) = 2m(G_0^{-1})_{gh}$.

The second term is given by

$$\begin{aligned} \tilde{\mu}^{(k)'}G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}\tilde{\mu}^{(k)} &= vec(\mathcal{U}^{(k)})'((G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}G_0^{-1})\otimes I_m)vec(\mathcal{U}^{(k)}) \\ &= \text{tr}\left((\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}G_0^{-1}\right) \\ &= \text{tr}\left(G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}\right) \end{aligned}$$

When g=h, $\tilde{\mu}^{(k)'}G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}\tilde{\mu}^{(k)} = (G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1})_{gg}$
When g $\neq$ h, $\tilde{\mu}^{(k)'}G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}\tilde{\mu}^{(k)} = 2(G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1})_{gh}$

The third term is given by

$$\begin{aligned} \text{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}V^{(k)}\right) &= \text{tr}\left((G_0^{-1}\otimes I_m)(\frac{\partial G_0}{\partial \gamma_{gh}}\otimes I_m)(G_0^{-1}\otimes I_m)V^{(k)}\right) \\ &= \text{tr}\left(\left((G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}G_0^{-1})\otimes I_m\right)V^{(k)}\right) \end{aligned}$$

Partition $V^{(k)}$ into 4 matrices of size m by m, i.e., $V^{(k)} = \begin{bmatrix} V_{11}^k & V_{12}^k \\ V_{12}^k & V_{22}^k \end{bmatrix}$.

When g=h,

$$\begin{aligned}(G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}G_0^{-1}) \otimes I_m &= \begin{bmatrix} \gamma^{1g}\gamma^{g1} & \gamma^{1g}\gamma^{g2} \\ \gamma^{2g}\gamma^{g1} & \gamma^{2g}\gamma^{g2} \end{bmatrix} \otimes I_m \\ &= \begin{bmatrix} \gamma^{1g}\gamma^{g1}I_m & \gamma^{1g}\gamma^{g2}I_m \\ \gamma^{2g}\gamma^{g1}I_m & \gamma^{2g}\gamma^{g2}I_m \end{bmatrix}.\end{aligned}$$

Then

$$\begin{aligned}\operatorname{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}V^{(k)}\right) &= \sum_{l=1}^{2}\sum_{s=1}^{2}\operatorname{tr}(\gamma^{sg}\gamma^{gl}I_m V_{ls}^{(k)}) \\ &= \sum_{l=1}^{2}\sum_{s=1}^{2}\gamma^{sg}\gamma^{gl}\operatorname{tr}(V_{ls}^{(k)}) \\ &= (G_0^{-1}\Psi G_0^{-1})_{gg}\end{aligned}$$

where $\Psi = \begin{bmatrix} \operatorname{tr}(V_{11}^{(k)}) & \operatorname{tr}(V_{12}^{(k)}) \\ \operatorname{tr}(V_{12}^{(k)}) & \operatorname{tr}(V_{22}^{(k)}) \end{bmatrix}$

When g $\neq$ h,

$$\begin{aligned}(G_0^{-1}\frac{\partial G_0}{\partial \gamma_{gh}}G_0^{-1}) \otimes I_m &= \begin{bmatrix} 2\gamma^{1g}\gamma^{h1} & 2\gamma^{1g}\gamma^{h2} \\ 2\gamma^{2g}\gamma^{h1} & 2\gamma^{2g}\gamma^{h2} \end{bmatrix} \otimes I_m \\ &= \begin{bmatrix} 2\gamma^{1g}\gamma^{h1}I_m & 2\gamma^{1g}\gamma^{h2}I_m \\ 2\gamma^{2g}\gamma^{h1}I_m & 2\gamma^{2g}\gamma^{h2}I_m \end{bmatrix}.\end{aligned}$$

Then

$$\operatorname{tr}\left(G^{-1}\frac{\partial G}{\partial \gamma_{gh}}G^{-1}V^{(k)}\right) = \sum_{l=1}^{2}\sum_{s=1}^{2}\operatorname{tr}(2\gamma^{sg}\gamma^{gl}I_m V_{ls}^{(k)}) = 2\sum_{l=1}^{2}\sum_{s=1}^{2}\gamma^{sg}\gamma^{gl}\operatorname{tr}(V_{ls}^{(k)}) = 2(G_0^{-1}\Psi G_0^{-1})_{gh}$$

Therefore the M-step for γ_{gh} can be written as

$$\frac{\partial E\left(l_2|\tilde{y}_2;\theta^{(k)}\right)}{\partial \gamma_{gh}} = \begin{cases} -\frac{1}{2}\left\{m(G_0^{-1})_{gg} - (G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1})_{gg} - (G_0^{-1}\Psi G_0^{-1})_{gg}\right\}, & g = h \\ -\frac{1}{2}\left\{2m(G_0^{-1})_{gh} - 2(G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1})_{gh} - 2(G_0^{-1}\Psi G_0^{-1})_{gh}\right\}, & g \neq h \end{cases}$$

Let above equation equal to zero results into following equation

$$m(G_0^{-1})_{gh} = (G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1})_{gh} + (G_0^{-1}\Psi G_0^{-1})_{gh}$$

and this can be written in matrix form as

$$m(G_0^{-1}) = (G_0^{-1}(\mathcal{U}^{(k)})'\mathcal{U}^{(k)}G_0^{-1}) + (G_0^{-1}\Psi G_0^{-1}).$$

Left and right Mutiplying by G_0 yields

$$m(G_0) = (\mathcal{U}^{(k)})'\mathcal{U}^{(k)} + \Psi$$

therefore,

$$\begin{aligned} m\gamma_{gh} &= ((\mathcal{U}^{(k)})'\mathcal{U}^{(k)})_{gh} + (\Psi)_{gh} \\ &= \tilde{\mu}_g^{(k)'}\tilde{\mu}_h^{(k)} + \mathrm{tr}(V_{gh}^k) \end{aligned}$$

and

$$\gamma_{gh} = \frac{1}{m}(\tilde{\mu}_g^{(k)'}\tilde{\mu}_h^{(k)} + \mathrm{tr}(V_{gh}^k)).$$

APPENDIX C. CODE

```
library(MASS)
library(Matrix)
library(nlme)
library(foreach)
library(doParallel)
library(openxlsx)
registerDoParallel(cores = detectCores())

comb <- function(...) {
  mapply('rbind', ..., SIMPLIFY=FALSE)
}
set.seed(628)
t_start <- proc.time()
GEM.out=NULL
nlme.out=NULL
lasso_out=NULL

m=100  # number of subject
####generate number of time points for each individual
####maximum number of time points is 10
tp_min=8
tp_max=10
tp <- round(runif(m,tp_min,tp_max))
age<- unlist(sapply(tp, function(x)
            c(1, sort(sample(2:tp_max, x-1, replace=FALSE)))))
ID=rep(1:m, times=tp)
n=length(age)
```

```
################true parameter values############
beta=c(3,3.6,8,-5.4,5,2,rep(0,34))
p=length(beta)  ## number of fixed effects(betas)
sig.u0=1.44
sig.u1<- 1 #variance of random effect u2
sig.u01<-0.64
sig.e<- 2  #variance of residual
rho=0.8
#############Generate design matrix corresponding to fixed effect#############
X1=runif(n, -0.3, 0.3)
X2=rnorm(n,X1, 0.5)
X3=runif(n,X2-1,X2+1)
X4=rnorm(n,0,2)
X567=mvrnorm(n,c(0,0,0), matrix(c(0.5,0.1,0.1,
                                  0.1,0.4,0.1,
                                   0.1,0.1,0.6),nrow=3), empirical = FALSE)
X_rest=mvrnorm(n,rep(0,31),diag(31))
X=cbind(rep(1,n),age,X1,X2,X3,X4,X567,X_rest)

##########split X into 100 individual design matrices######
X_list=split.data.frame(X,ID)

#################design matrix for random effect####################
age_list=split.default(age,ID)
Z0_list=split.default(rep(1,n),ID)
Z0=do.call(bdiag, Z0_list)
Z1=do.call(bdiag, age_list)
result <- foreach(re=1:100, .combine="comb", .multicombine=TRUE) %dopar%{
  G0=matrix(c(sig.u0,sig.u01,sig.u01,sig.u1),2,2)
  u=mvrnorm(m, rep(0, 2), G0, empirical=TRUE)
```

```
b0=length(u[,1])
b1=length(u[,2])
##simulate correlated error term with AR(1) structure and rho=0.8
Sigma=do.call(bdiag, lapply(age_list,
      FUN = function(x) rho^abs(outer(as.vector(x),as.vector(x),"-"))))
e=as.vector(t(mvrnorm(1, rep(0, n), sig.e*Sigma, empirical=FALSE)))

Z=cbind(Z0,Z1)
y=X%*%beta+Z0%*%u[,1]+Z1%*%u[,2]+e

####################ECM algorithm with one iteration of gradient decent#######
##initial value
iteration=0
sig.u0=1
sig.u1=0.25
sig.u01=0.35
sig.e=1.5
rho=0.5
cc=1                      ##initial convergence criterion value
alpha=0.0001 ##initial step size for projection gradient descent alogrithm
while (cc>1e-5) {
  iteration=iteration+1
  ##  step size for projection gradient descent alogrithm
  alpha=(2/(1000*(iteration+0.0005)))
  kapa=c(sig.u0, sig.u01,sig.u1,sig.e,rho)
  G0=diag(sig.u0,b0)
  G1=diag(sig.u1,b1)
  G01=diag(sig.u01,b0)
  G=rbind(cbind(G0,G01),cbind(G01,G1))
  Sigma=do.call(bdiag, lapply(age_list,
```

```
      FUN = function(x) rho^abs(outer(as.vector(x),as.vector(x),"-"))))
Sigma_list=lapply(age_list,
           FUN = function(x) rho^abs(outer(as.vector(x),as.vector(x),"-")))


if (rho==1){
SigmaInv_list=lapply(age_list,
FUN = function(x) rho^abs(outer(as.vector(x),as.vector(x),"-"))/length(x)^2)
X_Sinv_X=Reduce("+",lapply(X_list,
FUN = function(x) colSums(x) %*% t(colSums(x))/nrow(x)^2))
 } else if(rho==-1){
   SigmaInv_list=lapply(age_list,
   FUN = function(x) ginv(rho^abs(outer(as.vector(x),as.vector(x),"-"))))
   X_Sinv_X=Reduce("+",
   Map(function(x,y,z) t(x) %*% y %*% z, X_list, SigmaInv_list, X_list))
 } else{
   SigmaInv_list=lapply(age_list,
   FUN = function(x) solve(rho^abs(outer(as.vector(x),as.vector(x),"-"))))
   X_Sinv_X=Reduce("+",
   Map(function(x,y,z) t(x) %*% y %*% z, X_list, SigmaInv_list, X_list))
 }

###variance matrix for each individual H1=z%*%G%*%t(z)+sig.e*Sig###
Sigma_inv=do.call(bdiag, SigmaInv_list)

HInv_list=Map(function(x,y,z)
ginv(cbind(x,y)%*%matrix(c(sig.u0,sig.u01,sig.u01, sig.u1),2)
%*%t(cbind(x,y))+sig.e*z), Z0_list,age_list, Sigma_list)

H_inv=do.call(bdiag,HInv_list)
```

```
X_Hinv_X=Reduce("+",Map(function(x,y) t(x) %*% y %*% x, X_list, HInv_list))
Hinv_X=do.call(rbind,Map(function(x,y) y %*% x, X_list, HInv_list))
Sinv_X=do.call(rbind,Map(function(x,y) y %*% x, X_list, SigmaInv_list))

P=H_inv-Hinv_X%*%solve(X_Hinv_X)%*%t(Hinv_X)

U=Sigma_inv-Sinv_X%*%solve(X_Sinv_X)%*%t(Sinv_X)


V=G-G%*%t(Z)%*%P%*%Z%*%G
M=G%*%t(Z)%*%P%*%y
M0=M[1:m]  ##conditional expected value of u0
M1=M[(m+1):(2*m)]   ##conditional expected value of u1
V0=V[1:m,1:m] ##conditional variance of u0
V1=V[(m+1):(2*m),(m+1):(2*m)] ##conditional variance of u1
V01=V[1:m,(m+1):(2*m)]


###update for sig.e#####
sig.e_new=drop(t(y-Z%*%M)%*%U%*%(y-Z%*%M)+sum(diag(t(Z)%*%U%*%Z%*%V)))/(n-p)

##use projected gradient decent method to update rho
##first derivative of Sigma w.r.t rho
Sigma_d=do.call(bdiag, lapply(age_list, FUN = function(x)
abs(outer(as.vector(x),as.vector(x),"-"))*
rho^abs(outer(as.vector(x),as.vector(x),"-"))-1))

##first derivative of negative E(l_1 | y_2; theta) w.r.t rho
grad_rho=1/2*(sum(diag(U%*%Sigma_d))
        -(1/sig.e_new)*drop(t(y-Z%*%M)%*%U%*%Sigma_d%*%U%*%(y-Z%*%M)
```

```
          +sum(diag(t(Z)%*%U%*%Sigma_d%*%U%*%Z%*%V))))

  g_d=rho-alpha*grad_rho ##gradient decent update
  rho_new=sign(g_d)*min(1, abs(g_d)) ##projected gradient decent update
  sig.u0_new=drop(t(M0)%*%M0+sum(diag(V0)))/b0
  sig.u1_new=drop(t(M1)%*%M1+sum(diag(V1)))/b1
  sig.u01_new=drop(t(M0)%*%M1+sum(diag(V01)))/b1

  kapa_new=c(sig.u0_new,sig.u01_new,sig.u1_new,sig.e_new,rho_new)
  cc=sqrt(sum((kapa_new-kapa)^2)/sum((kapa_new)^2))   #converge criteria#
  sig.u0=sig.u0_new
  sig.u1=sig.u1_new
  sig.e=sig.e_new
  sig.u01=sig.u01_new
  rho=rho_new
}
GEM.out=rbind(GEM.out,c(sig.u0,sig.u01,sig.u1,sig.e,rho,iteration))
Y=as.vector(y)
data=data.frame(ID,Y,age,X1,X2,X3,X4,X567,X_rest)
out=try(lme(Y~age+X1+X2+X3+X4+X1.1+X2.1+X3.1+X1.2+X2.2+X3.2+X4.1+X5+X6+X7+X8+X9
    +X10+X11+X12+X13+X14+X15+X16+X17+X18+X19+X20+X21+X22+X23+X24+X25+X26
    +X27+X28+X29+X30+X31, data=data,random=~age|ID,
   correlation = corExp(form = ~age |ID)),silent=TRUE)
   if(grepl( "Error", out[1], fixed = TRUE)){
   nlme_out1=c("NA","NA","NA","NA","NA")} else{
   phi=as.vector(exp(-1/coef(out $modelStruct$corStruct,unconstrained=FALSE)))
   nlme_out1=c(VarCorr(out)[1,1],VarCorr(out)[2,3],VarCorr(out)[2,1],
               VarCorr(out)[3,1], phi)}
   nlme.out=rbind(nlme.out,nlme_out1)
```

```
################################################
#####variable selection by LASSO#############
################################################
G0=diag(sig.u0,b0)
G1=diag(sig.u1,b1)
G01=diag(sig.u01,b0)
G=rbind(cbind(G0,G01),cbind(G01,G1))
Sigma=do.call(bdiag, lapply(age_list,
     FUN = function(x) rho^abs(outer(as.vector(x),as.vector(x),"-"))))
R=sig.e*Sigma  ##residual variance##
H=Z%*%G%*%t(Z)+R
if (abs(rho)!=1){
    D=chol(round(solve(H),6))} else {
    D=chol(round(ginv(as.matrix(H)),6))}
X_star=D%*%X
Y_star=D%*%y
beta=rep(0,p)
LASSO=NULL
lambda_max=80/log(n)/sqrt(n)
lambda_min=20/log(n)/sqrt(n)
grid <- exp(seq(log(lambda_max),log(lambda_min),length = 1000))
for (lambda in grid){
  tol=1
  while(tol>0.00006){
  beta_new=beta
  beta[1]=as.numeric(t(X_star[,1])%*%(Y_star-X_star[,-1]%*%beta[-1])
                   /t(X_star[,1])%*%X_star[,1])
  for (k in 2:p) { beta1=as.numeric(t(X_star[,k])%*%(Y_star-X_star[,-k]
                       %*%beta[-k]))/t(X_star[,k])%*%X_star[,k]
```

```
                    lambda1=n*lambda/(2*t(X_star[,k])%*%X_star[,k])
                   if (beta1 > lambda1){
                        beta[k]=beta1-lambda1
                     } else if (beta1 < -lambda1){
                        beta[k]=beta1+lambda1
                       } else{beta[k]=0}
                   }
     tol=sqrt(sum((beta_new-beta)^2))
     }
   BIC=(sum(Y_star-X_star%*%beta)^2/n)+sqrt(2)*log(n)*sum(beta!=0)/sqrt(n)
   lasso=cbind(lambda,BIC,t(beta))
   LASSO=rbind(LASSO,lasso)
   }
   lasso_best=LASSO[which(LASSO[,2]==min((LASSO[,2]))),]
   lasso_out=rbind(lasso_out,lasso_best)
   #lasso_out=rbind(lasso_out,lasso)

 list(GEM.out,nlme.out,lasso_out)

}
```