R mlmmm Package: Fitting Multivariate Linear Mixed Effects Models with Missing Values

R mlmmm Paketi: Kayıp Veriler Olduğunda Çok Değişkenli Doğrusal Karma Etkili Modellerin Uydurulması

Recai M. YÜCEL[a]

[a]Department of Epidemiology and Biostatistics,
School of Public Health,
State University of New York at Albany, One University Place,
Rensselaer, NY

Geliş Tarihi/*Received:* 20.10.2014
Kabul Tarihi/*Accepted:* 24.11.2014

Yazışma Adresi/*Correspondence:*
Recai M. YÜCEL
Department of Epidemiology and Biostatistics,
School of Public Health,
State University of New York at Albany, One University Place,
Rensselaer, NY
ryucel@albany.edu

ABSTRACT Objective: Publicly available R package **mlmmm** is presented. **mlmmm** implements an efficient estimation technique based on a hybrid of EM and Fisher scoring algorithm, operating under a multivariate generalization of a well-known linear mixed-effects model in the presence of incompletely observed grouped data. **Material and Methods: mlmmm** is specifically designed for grouped data which are commonly encountered in longitudinal studies where the repeated observations are grouped within the individuals or in clustered sample surveys. The analyses of such data are often more complicated by the inevitable missing data. **Results:** When the goal of the analysis to model multiple characteristics jointly using incompletely-observed data, **mlmmm** can be used to obtain maximum likelihood estimates of desired parameters, test variance components or profile clusters using their empirical Bayes estimates. Some of the estimation procedure is based on efficient model-fitting technique developed by Schafer and Yucel 2002. **Conclusion:** The algorithms presented here provide excellent convergence and reliability. This behavior is best realized when the missing data and number of response variables are in moderately low and when the auxiliary variables are informative not only in their predictiveness but also in their ability in explaining missingness in the response variables, making the the underlying MAR assumption more plausible.

Key Words: Missing data; MLE; EM and Fisher-scoring;
mixed-effects; longitudinal data; clustered data

ÖZET Amaç: Halka açık R paketi **mlmmm** sunulmuştur. **mlmmm**, eksik olarak gözlenen gruplanmış verinin olması durumunda kullanıldığı bilinen doğrusal karma-etkili modellerin çok değişkenli genelleştirilmiş hali üzerine çalışan, EM ve Fisher skorlama algoritmasının hibridi üzerine temellenen etkin bir tahmin tekniği sağlamaktadır. **Gereç ve Yöntemler: mlmmm**, tekrarlı gözlemlerin birimler içinde ya da kümelenmiş örnekleme çalışmalarında gruplandığı uzunlamasına çalışmalarda genellikle karşılaşılan gruplanmış veriler için özel olarak tasarlanmıştır. Bu tip verilerin analizleri, kaçınılmaz kayıp veriler nedeniyle genellikle daha da zor bir hale gelir. **Bulgular:** Analizin amacı, eksik olarak gözlenen veriler kullanarak çoklu özellikleri birleşik olarak modellemek olduğunda, istenen parametrelerin en çok olabilirlik tahmin edicilerinin, test varyans bileşenlerinin veya deneysel Bayes tahminlerini kullanarak profil kümelerin elde edilmesi için **mlmmm** kullanılabilir. Bazı tahmin yöntemleri Schafer and Yucel 2002 tarafından geliştirilen etkin model-uyum tekniklerine dayanmaktadır. **Sonuç:** Burada sunulan algoritmalar mükemmler bir yakınsama ve güvenilirlik sağlamaktadır. Bu davranış, kayıp veri ve yanıt değişkeni sayısı kısmen düşük olduğunda ve yardımcı değişkenler sadece kendi tahmin edilebilirlikleri yönünden değil, aynı zamanda yanıt değişkenlerindeki kayıp veriyi açıklama kabiliyetleri yönünden de aydınlatıcı olduklarında en iyi şekilde ortaya çıkmaktadır.

Anahtar Kelimeler: Kayıp veriler; en çok olabilirlik tahmin edicisi; EM ve Fisher-skorlama;
karma-etkiler; uzunlamasına veriler; kümelenmiş veriler

doi: 10.5336/biostatic.2014-42149

Turkiye Klinikleri J Biostat 2015;7(1):11-24

BACKGROUND

Presence of missing data is a highly prevalent problem in most applications. In applications with grouped data (e.g. longitudinal or clustered data), handling missing values in a principled manner is an important task. Because most statistical methods and their implementations are not specifically designed to handle missing values, practitioners are often faced with making undesired choices such as case deletion, last observation carried forward (in longitudinal applications) which can do serious harm to the inferences such as biased estimates, artificially increased precision. In designs resulting in grouped data (e.g. longitudinal or clustered data), such ad-hoc methods can potentially be very damaging to the inferences as important relationships among incompletely-observed variables and others as well as distinct variation sources (within-cluster and between-cluster) are essentially ignored.

Since the seminal work by Dempster et al.[1] on model-fitting algorithms designed for missing data problems or problems that can be formulated as missing data, there has been many important developments in the statistical theory underlying missing data methods (e.g. Rubin,[2] Rubin,[3] Little and Rubin[4]). Advancement in computational statistics has helped the dissemination of these developments via software products. A particularly popular techniques are model-based techniques where inferences are produced by postulating models for the observed data and basing the inferences on the implied likelihood or posterior distribution. Two of the most preferred method of inference are maximum likelihood (ML) inference (or inference based on posterior modes)[1,2,5] and multiple imputation (MI) inference.[2,5] Computational methods such as expectation maximization (EM), Fisher scoring are often used to obtain ML estimates under the postulated observed data likelihood. MI inference consists of two steps, replacing missing values by a moderate number of plausible values typically drawn from the predictive distribution of missing data given observed data and combining estimates and standard estimates obtained from the analyses of "completed or imputed data". The choice of the method often depends on several criteria of which the most important is the availability of software and practicality from the practitioners' point of view.

Previous methodology on missing data and its implementation in software products have mostly focused on the data problems where the observations are independently drawn from identical distributions. Some examples include a key book by Schafer[5] providing a thorough treatment of missing data techniques in widely encountered problems, which are later implemented in several software packages including Splus missing library, R packages **norm**, **cat** and **mix**.[*,6] These packages implement algorithms for finding ML estimates as well as creating multiple imputations of missing values for continuous data (operating under a multivariate normal model), categorical data (operating under a loglinear model) and mixed data (under a general location model), respectively. A useful alternative approach based on variable-by-variable approach was developed by Raghunathan et al. for creating multiple imputations to conduct inferences in surveys which typically more prone to missing data and other complications such as skip patterns, bounds, etc.[7] This method is available as a SAS-callable macro IveWare; a similar approach has also been developed by Buuren and Groothuis-Oudshoorn[8] and implemented in R package called **mice**. Another commonly used set of procedures for the analysis of *ungrouped data* was implemented in a SAS procedure called SAS PROC MI.[9]

As the design used to draw observations becomes complicated, sound statistical technique specifially designed to handle missing values also become complicated. In longitudinal studies or clustered data, for example, statistical methods for handling missing values should take into account of important relationships such as variation within and between subjects and clusters. The studies on this topic have so far focused on creating multiple imputations. Schafer and Yucel;[10] Liu, Taylor and Belin[11] develop methods on creating multiple imputations in correlated data. The multiple imputation framework for grouped data as described by Schafer and Yucel[10] has also been implemented in R package **pan**.

---

[*] R. R: A Language and Environment for Statistical Computing. Vienna: R Foundation for Statistical Computing; 2011. URL http://www.R-project.org/. ISBN 3-900051-07-0.

Recently, multilevel modeling software MlWin using primarily Markov Chain Monte Carlo (MCMC) techniques has also developed routines for multiple imputation inference.[12] Advantages of multiple imputations are reviewed by many including Rubin,[3] Schafer.[5] From a practitioner point of view, because these methods are mostly simulation-based iterative techniques, a great care should be taken for monitoring convergence, especially in methods based on variable-by-variable approaches. For this purpose, methods such as EM or Fisher scoring could be beneficial for their stability (i.e. increasing the likelihood) and speedy convergence.

R mlmmm PACKAGE

This paper introduces **mlmmm** package within a publicly available R language and software environment, to fit multivariate linear mixed-effects models in the presence of missing values.[*,10,11] These models are often utilized to simultaneously model several continuous characteristics observed over time for a group of individuals or for a group of individuals nested within units (e.g. patients within doctors). Examples where these models may be desired include joint modeling of substance use characteristics such as nicotine level in saliva and alcohol level over time as a function of individual characteristics and intervention; in clustered samplings where the schools are sampled at the first stage followed by a selection of students, a joint modeling of test scores might be of interest. In general, **mlmmm** is used to fit a joint model of incompletely-observed multiple characteristics, allowing means and covariances to vary across the clusters.

There have been other similar models and techniques for model fitting under the similar models, however, the development of these models has not been equal to its univariate counterparts. There is vast literature on the mixed-effects models starting with studies by Laird and Ware,[13] Jennrich and Schluchter.[14] With the computational advances, many software products have been available to the consumers including HLM, SAS, R packages **lme** and **lme4**. The multivariate extensions have been studied by Liu, Taylor and Belin[11] by means of MCMC, Shah, Laird and Schoenfeld[15] who developed a conventional EM algorithm for bivariate case. Schafer and Yucel developed a fast MLE algorithm based on a hybrid of EM and Fisher scoring algorithm, which was used in conjunction with MCMC algorithm for multiple imputation inference.[10] The focus of the current paper is to provide computational principles of this hybrid EM algorithm as implemented in the R package called **mlmmm**.

Installation of mlmmm proceeds in the same fashion as other R packages using *install.packages("mlmmm")*. There are a total of three commands within this package: *mlmmm.em()* fits a multivariate linear mixed-effects model with unstructured covariance matrix on the random-effects covariance matrix; *mlmmmbd.em()* fits a multivariate linear mixed-effects model with block-diagonal covariance matrix on the random-effects covariance matrix; and *mlm.em()* fits multivariate linear regression model with missing response values (fixed-effects only version). An example script can be run by the following command: *example(mlmmm)* after executing *library(mlmmm)*. There are two datasets that come with the **mlmmm**: *data(adg)* and *data(pain)*. These datasets are described in detail in Sections 3 and 4, and associated help files are also available within **mlmmm**.

The rest of this paper is organized as follows. Section 2 reviews the multivariate mixed-effects model and gives the computational algorithm implemented in **mlmmm** package. Section 3 and 4 provide examples illustrating **mlmmm** on two examples with clustered and longitudinal data.

DATA, MODEL AND METHODS FOR MODEL FITTING

MULTIVARIATE GROUPED DATA WITH MISSING VALUES

Longitudinal studies or studies conducted under clustered sampling are subject to grouped data. Multivariate longitudinal or clustered data occur when multiple responses measured over time for a group of subjects (possibly at different occasions for each individual), or, multiple responses measured

---

[*] R. R: A Language and Environment for Statistical Computing. Vienna: R Foundation for Statistical Computing; 2011. URL http://www.R-project.org/. ISBN 3-900051-07-0.

for subjects nested within naturally occurring groups. Examples include multi-stage sampling where individuals are nested within a primary sampling unit such as a county or a school; clinical studies where individuals are observed for several characteristics over time. Highly prevalent problem in such studies is missing data, and the underlying model estimated by **mlmmm** is based on a multivariate extension of a popular linear mixed-effects model for longitudinal or clustered data with arbitrary patterns of missing values. Figures 1 and 2 depict the examples of longitudinal or clustered data with missing values, respectively. **mlmmm** assumes that mechanism producing these missing values is missing at random in the sense defined by Rubin[2] and Little and Rubin.[4]

MULTIVARIATE LINEAR MIXED-EFFECTS MODEL (MLMM)

Each of the responses matrices depicted in Figures 1 and 2 is denoted by $(n_i \times r)$ for sample unit i ; $i = 1, 2 \ldots, m$. Note that each row of y_i is a joint realization of variables $Y_1, Y_2, \ldots, Y_r$. Multivariate linear mixed-effects model is given by

$$y_i = X_i\beta + Z_i b_i + \in_i \qquad (1)$$

where X_i (n_i x p) and Z_i (n_i x q) are known covariate matrices, β $(p \times r)$ is a matrix of regression coefficients common to all units, and b_i (q x r) is a matrix of coefficients specific to unit i. In popular terminology, β and b_i are called "fixed effects" and "random effects", respectively. The n_i rows of $\in_i$ are assumed to be independently distributed as $N(0; \sum)$, and that the random effects are distributed as $\mathrm{vec}(b_i) \sim \mathrm{N}(0;\Psi)$ independently for $i = 1, \ldots, m$ (the "vec" operator vectorizes a matrix by stacking its columns). Alternatively, the model (1) can be represented as

$$vec(y_i) \sim N_{n_i r}(vec(X_i\beta), (I_r \otimes Z_i)\, \Psi\, (I_r \otimes Z_i)^T + \sum \otimes I_{n_i}) \qquad (2)$$

In longitudinal applications, the matrices X_i and Z_i may include times of measurements, allowing relevant aspects of the growth to vary across

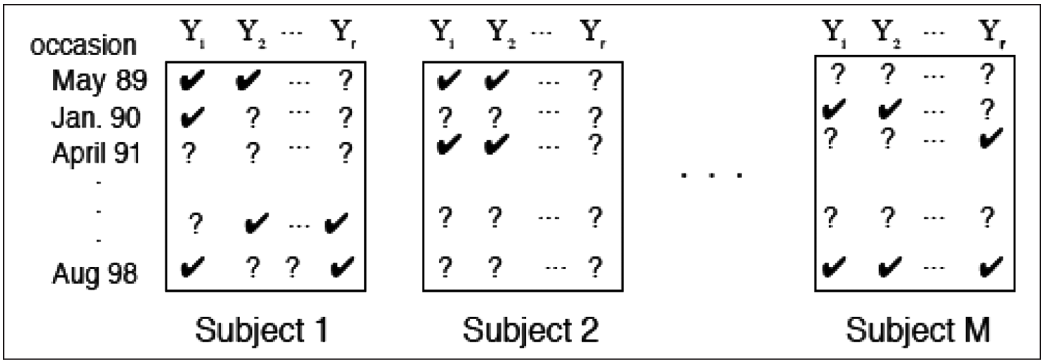

FIGURE 1: Multivariate longitudinal data with missing values denoted by question marks.

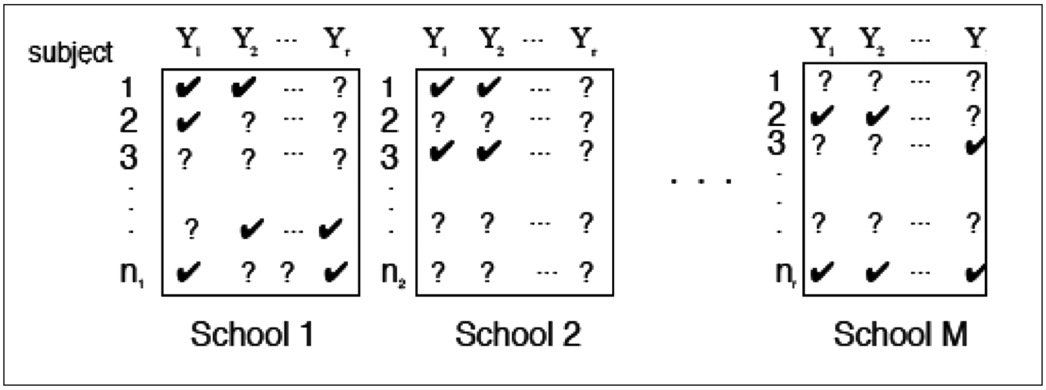

FIGURE 2: Multivariate clustered data with missing values denoted by question Marks.

subject. Similarly, in clustered applications, cluster-specific or individual-specific covariates may be included in X_i or Z_i. In applications where the data are drawn under a cluster sampling, for example, inclusion of an intercept in Z_i may suffice to capture the variations at both levels of clusters and individuals.

mlmmm allows two structures on Ψ, the underlying covariance matrix for the random-effects $b_1, b_2, \ldots, b_m$: unstructured and block-diagonal. Unstructured Ψ allows the random effects for any two responses Y_i and Y_k to be correlated, and block-diagonal Ψ assumes that the random effects for each response are independent of those for any other response. Note that block-diagonal assumption is a priori assumption and relies on a previous knowledge on the substantive problem. The choice also relies on estimability; in clustered data applications, for example, estimation of the unstructured Ψ may not be feasible if the number of responses is large.

MODEL FITTING TECHNIQUE

mlmmm implements a model fitting method that combines the speed of the Fisher's scoring algorithm and the stability and consistency of EM algorithm.[1] Below key points of the computational algorithm are provided, I refer readers to Schafer and Yucel for details.[10]

In our settings, a traditional EM algorithm regards the missing data as well as random effects as missing data, and bases its algorithmic details on the following factorization of the augmented-data likelihood:

$$L(\theta \mid B) = L(\Psi \mid \mathrm{B})\mathrm{L}(\beta,\textstyle\sum \mid \mathrm{Y},\mathrm{B}) \tag{3}$$

where $\theta = (\Psi,\beta,\sum)$, $Y = (Y_1, \ldots, Y_m)$ and $B = (b_1, \ldots, b_m)$. The overall maximum of the above likelihood with respect to θ can be found by maximizing each of the two factors separately, neither of which requires iteration. Each cycle of EM maximizes the expected logarithm of (3), where the expectation is taken with respect to the conditional distribution of B given Y with the parameters fixed at their current estimates. However, this conventional EM algorithm may suffer from very slow convergence, and this slow convergence may become worse as the missing data portions become large.

Complete-Data Procedure

When there are no missing values in y_i, Fisher scoring algorithm can be a better and more computationally efficient method to maximize the following complete-data likelihood function,

$$L(\theta) \propto \textstyle\prod_{i=1}^{m} \left| (I_r \otimes Z_i)\Psi(I_r \otimes Z_i)^T + \sum \otimes I_{n_i} \right|^{-1/2} exp\left\{-\tfrac{1}{2}\delta_i^T((I_r \otimes Z_i)\Psi(I_r \otimes Z_i)^T + \textstyle\sum \otimes I_{n_i})^{-1}\delta_i\right\},$$

where $\delta_i = \mathrm{vec}(y_i - X_i\beta)$. Unlike (3), the likelihood function given above does not include the random-effects, and by working with this likelihood function, **mlmmm** does not suffer from slow convergence as the amount of "missing data" is now reduced.

Note that the variance-covariance matrix above requires inversion and finding the de-terminant of a large matrix whose dimensions depend on the number of observations, to overcome this problem, we use the following relationship:

$$|W_i| = \left|(I_r \otimes Z_i)\Psi(I_r \otimes Z_i)^T + \textstyle\sum \otimes I_{n_i})^{-1}\right| = \left|\textstyle\sum \otimes I_{n_i}\right|^{-1}|\Psi|^{-1}|U_i|,$$

where

$$U_i = (\Psi^{-1} + (\textstyle\sum^{-1} Z_i^T \otimes Z_i))^{-1}.$$

The simplified version of the log-likelihood is,

$$\ell(\theta) = -\frac{N}{2}log|\textstyle\sum| - \frac{m}{2}log|\Psi| + \frac{1}{2}\sum_{i=1}^{m} log|U_i| - \frac{1}{2}\sum_{i=1}^{m} \delta_i^T W_i \delta_i \tag{4}$$

Fisher scoring updates the current estimate θ^{t+1} by solving linear system $C\,\theta^{t+1} = d$, where $C = -E(\ell''(\theta^{(t)}))$ and $d = C(\theta^{(t)} + \ell'(\theta^{(t)})$. Upon convergence, the final value of C^{-1} provides an estimated covariance matrix for $\hat{\theta}$.

For convenience of calculating respective derivative, **mlmmm** operates on β and the nonredundant elements of Ψ^{-1} and $\sum^{-1}$. These matrices can be expressed as

$$\Psi^{-1} = \textstyle\sum_{j=1}^{g} \omega_j G_j,$$

$$\textstyle\sum^{-1} = \sum_{j=1}^{h} \sigma_j F_j,$$

where $G_1, G_2, ..., G_g$ and $F_1, F_2, ..., F_h$ are known symmetric matrices of dimensions $rq \times rq$ and $r \times r$, respectively. The number of free parameters in Ψ is $g = rq(rq + 1) / 2$ when Ψ is unstructured and $g = rq(rq + 1) / 2$ when it is block-diagonal.

Incomlete-Data Procedure

mlmmm combines a traditional EM with Fisher scoring on as explained above to overcome slow convergence when portions of $Y = (y_1, y_2, ..., y_m)$ are ignorably missing.[2,4] In other words, Fisher scoring is embedded within an EM algorithm which augments the observed data with missing portions of y_i but not random effects. As in many missing-data algorithms, the performance of this algorithm is best when the proportion of partially observed rows in y_i is small, and degrades if the observed data become very sparse; however, it does not tend to slow down merely when the random effects are poorly estimated.

Some of the noteworthy points of the algorithm for the users of **mlmmm** are given below. The loglikelihood function given in (4) is a linear function of the sufficient statistics vec(y_i) and vec(y_i)$vec(y_i)^T$. The expectations of these two statistics given the observed data are calculated as follows:

$$E\left(y^*_{ij(mis)} \middle| y_{i(obs)}, b_i\right) = \sum\nolimits_{21} \sum\nolimits_{11}^{-1} y^*_{ij(obs)}$$

where $\sum_{11}$ is the square sub-matrix of $\sum$ corresponding to the observed elements, $\sum_{21}$ is the rectangular sub-matrix of covariances between the missing and observed elements, and $y^*_{ij(mis)}$ and $y^*_{ij(obs)}$ denote the missing and observed portions of the jth row of $y^*_i = y_i - X_i\beta - Z_i b_i$. Because y^*_i is a linear function of b_i, the expectation of y_i without conditioning on b_i is obtained by direct substitution of $E(b_i|y_{i(obs)}$ for b_i. Notice that the value of $\sum_{21}\sum_{11}$ varies by missingness pattern but not by observational units $i = 1,2,...,m$; computations can be reduced by grouping rows with identical missingness patterns across units. The parameters of the distribution of b_i given $y_{i(obs)}$ are obtained by applying a reverse-sweep procedure to $\hat{b}_i$ and U_i to de-condition upon $y_{i(mis)}$.

For the second sufficient statistic vec(y_i) $vec(y_i)^T$, one can apply a similar argument, first calculating the conditional expectation given b_i and $y_{i(obs)}$, then averaging over the distribution of b_i given $y_{i(obs)}$. Let y_{ijk} denote the kth element of the jth row of y_i. The formula for the expectation of $y_{ijk}y_{ij'k'}$ depends on whether y_{ijk} and $y_{ij'k'}$ are observed or missing, and whether they are in the same $(j = j')$ or different $(j \neq j')$ rows. It is easy to see that the expectation of $y_{ijk}y_{ij'k'}$ given $y_{i(obs)}$ is given by: $y_{ijk}y_{ij'k'}$ if both are observed; y_{ijk} E($y_{ij'k'}|y_{i(obs)}$) if y_{ijk} is observed and $y_{ijk}y_{ij'k'}$ is missing; and

$$E\left(y_{ijk}\middle|y_{i(obs)}\right)E\left(y_{ij'k'}\middle|y_{i(obs)}\right) + Cov\left(y_{ijk}, y_{ij'k'}\middle|y_{i(obs)}\right)$$

if both are missing. The covariance between y_{ijk} and $y_{ij'k'}$ given $y_{i(obs)}$ is equal to

$$Cov\left(A_{ijk}, A_{ij'k'}\middle|y_{i(obs)}\right) + \left[\sum\nolimits_{22.1}\right]_{kk'}$$

if they are in the same row, and

$$Cov\left(A_{ijk}, A_{ij'k'}\middle|y_{i(obs)}\right)$$

if they are in different rows, where

$$A_{ijk} = E\left(y_{ijk}\middle|b_i, y_{i(obs)}\right)$$

comes from the regression predictions for the missing elements in the jth row of y_i given the observed elements. The covariance $Cov\left(A_{ijk}, A_{ij'k'}\middle|y_{i(obs)}\right)$ is obtained by noting that it is a linear function of the elements of the covariance matrix for b_i given $y_{i(obs)}$.

The M-step requires us to maximize the expected loglikelihood computed in the E-step. This expected loglikelihood has nearly the same form as (4) and can be maximized by a slight modification of the Fisher scoring procedure.

Although the model fitting in the presence of missing data in **mlmmm** is speedier than conventional EM methods, it may be less stable when the loglikelihood is oddly shaped. To improve stability, **mlmmm** also implements a traditional EM procedure based on the augmented likelihood (3). When the scoring fails to increase the loglikelihood, the algorithm replaces our EM with the traditional EM step to guarantee the loglikelihood increases.

By-products of these algorithms that are available in **mlmmm** includes the followings:

- *Standard errors of β*: It is well known that conventional EM algorithm does not produce the standard errors of the unknown parameters. Because the Fisher scoring algorithm is used in the M-step, upon convergence, expected information matrix is calculated to produce the standard errors.
- *Likelihood values*: The likelihood values are also calculated at each iteration, which can also be used to assess the goodness-of-fit of the models under investigation.
- *Estimates of the random-effects*: Empirical Bayes estimates of the random-effects are available, which can be used in applications where the profiling of individual clusters is required.
- *Missingness patterns*: Functions of **mlmmm** provide missingness patterns observed in y. See examples for detailed usage.

FURTHER NOTES

Algorithms designed to maximize likelihood functions resulting from complex incompletedata problems, especially scoring, may require good starting values. **mlmmm**, by default, obtains starting values as follows: For each response variable Y_j, a univariate linear mixed model using the cases for which Y_j is observed. Fast and stable algorithms implemented in **lmm** provide ML estimates for the portions of $\sum$, Ψ and β pertaining to Y_j. Off-diagonal elements of $\sum$ and blocks of are initially set to zero.

Sections 3 and 4 below demonstrate the details of examples of **mlmmm**.

EXAMPLE 1:
CLUSTERED DATA APPLICATION

Data example used here come from “SAS System for Mixed Models” by Littell et al.[16] and it is an example data within **PROC MIXED** documentation. This dataset was collected to investigate average daily gains (AGD) of steers fed for 160 days. Treatments are four diets consisting of a base ration and three levels of a medicated feed additive added to the base ration. Objective of the experiment was to determine the optimal level of feed additive to maximize the average daily gain. Total of 32 steers were housed in eight barns, the blocking factor, where each barn held four steers which were individually fed. Note that barns give rise to correlated observational units, steers. Therefore, in the setting below, we will view the barn as the clustering unit. Steers were randomly assigned to treatments. For the purposes of illustration we imposed missing values on the response values under missing completely at random fashion.

SET-UP

As any model-fitting procedure, functions of **mlmmm** require assignments of response matrix Y where the missing values should be denoted as “NA”, predictor or design matrix X, specific columns of X to indicate the fixed and the random-effects are required and a vector indicating the cluster number of observations. Here the response variables are average daily gain in grams (add$y.1), initial weight on the log scale (adg$y.2).

```
library(mlmmm)

data(adg)

y=cbind(adg$y.1,adg$y.2)#form the response matrix colnames (y)=c(“adg”, “initwt”)

library(lattice)

xyplot(y[,1]~log(y[,2] | subj,ylab=”Average Daily Gain”, xlab=”Initial Weight”, data=adg)
```

Figure 3 displays the relationship of daily gain versus initial weight in each barn. To extent sample size allows for speculation, average daily gain seems to be positively correlated with the initial weight in barns 1, 3 and 8; no apparent relationship is seen in barns 2, 4, 5, and 7; and small negative correlation in barn 6. The visual evidence motivates the use of barn-specific effects, or random-effects. However, the small number of observations per cluster limits the random-effects to only intercept (i.e. $Z_i = 1$).

A vector, *subj*, indicating the cluster membership needs to be specified to use the functions of **mlmmm**:

```
subj=adg$subj # subj is barn
```

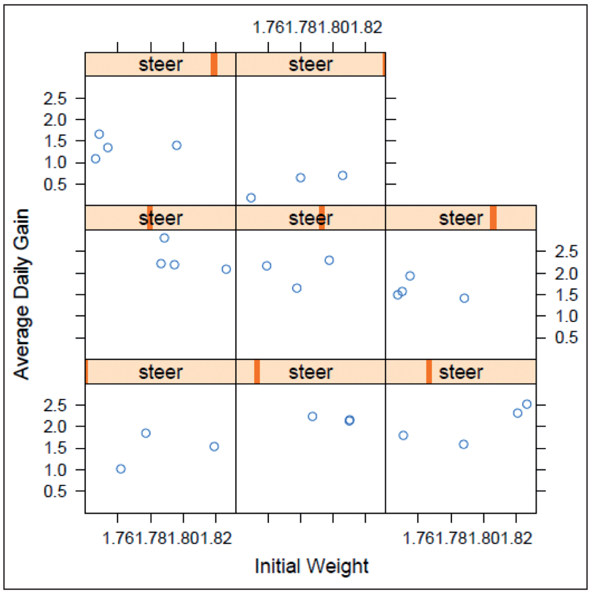

FIGURE 3: Relationship of initial weight and daily gain before and after deletion.

Now we must specify the design matrix, or X. Because the four treatment are not clearly ordered, we choose X with columns corresponding to an intercept and three dummy covariates for treatments. This particular selection of X allows the population means for the four subjects to be estimated freely. We will also allow the intercept to randomly vary by subject. When using the functions (*mlmmm.em(), mlmmmbd.em(), mlmm.em()*), the individual covariate matrices are combined into a single matrix called *pred*. The pred matrix has the same number of rows as Y. Each column of X_i and Z_i must be represented in pred. In this example, because Z_i is merely the first column of X_i, we do not need to enter that column twice. Finally, *pred* is simply the matrix X_i, stacked upon itself eight times, which corresponds to number of barns:

```
# dummies below are treatment indicators
pred<cbind(adg$pred.int,adg$pred.dummy1,adg$pred.dummy2, adg$pred.dummy3)
xcol<-1:4
zcol<-1
###### Brief look at the response, subj and pred:
> head(y)
   adg  Initwt
[1,] 1.03 5.823046
[2,] 1.54 6.167516
[3,] NA   6.095825
[4,] 1.86 5.913503
[5,] 1.31       NA
[6,] 2.16 6.111467
> head(subj)
># the first rows of y belong to barn 1, second four belong to barn 2, etc. :
[1] 1 1 1 1 2 2
> head(pred)
   int trt1 trt2 trt3
[1,] 1   1    0   0
[2,] 1   0    1   0
[3,] 1   0    0   1
[4,] 1   0    0   0
[5,] 1   1    0   0
[6,] 1   0    1   0
```

The setup of the arguments required by the functions of **mlmmm** is now complete. Next section will focus on the three functions: *mlmmm.em(), mlmmmbd.em()* and *mlm.em*:

MODEL FITTING VIA THE FUNCTIONS OF MLMMM

Here we fit three different models: unstructured Ψ, block-diagonal Ψ and finally fixed-effects only model.

```
# Model 1, unstructured Psi
# below the argument maxits indicate the maximum number of iterations and
# eps indicates the convergence criteria
unst.psi.result<mlmmm.em(y,subj,pred,xcol,zcol,maxits=200,eps=0.0001)
# results:
# estimates of the fixed-effects:
> unst.psi.result$beta
        [,1]       [,2]
[1,] 1.80250000  5.87971447
[2,] -0.46625000 0.06618314
[3,] 0.06625000  0.12094579
[4,] -0.09433734 0.08992500
```

```
#standard errors of these estimates, s.e.(vec(\hat{beta})):
> sqrt(diag(unst.psi.result$xtwxinv))
[1] 0.20734241 0.11940668 0.11940668 0.11940668
0.05418227
[6] 0.07126197 0.07126197 0.07126197
# estimates of the variance components:
> unst.psi.result$sigma
        [,1]      [,2]
[1,] 0.05703166 0.01981654
[2,] 0.01981654 0.02031310
> unst.psi.result$psi
        [,1]       [,2]
[1,] 0.28689583 0.027679844
[2,] 0.02767984 0.003172657
# likelihood values at each iteration:
> unst.psi.result$$logll
[1] 54.44853 59.08997 65.07784 67.99571 68.81427
69.15461 [7] 69.21332 69.22813 69.23204 69.23318
69.23353 69.23364 [13]69.23368 69.23369
```

Investigating the fixed effects β and its standard errors indicate that only treatment 1 effect is significantly associated with daily gain. For treatments 1 and 3 this is a negative association, and the estimates show that there is a decrease of -.46625 and -.09433 units in these gains for treatments 2 and 3. Initial weight (on the log scale) is not significantly associated with the treatments, which, substantively, provides an opportunity to see the effects that are purely attributable to the treatment rather that initial weight. Note, however, that the size of clusters can be a barrier in obtaining consistent estimates and presents a challenge in this example, and these results should be interpreted with this in mind. To overcome the challenge of estimation, one might consider eliminating a priori assumed dependency on the random-effects b_1 and b_2. Let's see how the model assuming independent random-effects t in this example:

```
# Model 2, block-diagonal Psi
# below the argument maxits indicate the maximum
number of iterations and
# eps indicates the convergence criteria
>bd.psi.result<mlmmmbd.em(y,subj,pred,xcol,zcol,
maxits=200,eps=0.0001)
# results:
# estimates of the fixed-effects:
> bd.psi.result$beta
        [,1]       [,2]
[1,] 1.80250000  5.87971447
[2,] -0.46625000 0.06463376
[3,] 0.06625000  0.12094579
[4,] -0.09847355 0.08992500
#standard errors of these estimates, s.e.(vec(\hat{beta})):
> sqrt(diag(bd.psi.result$xtwxinv))
[1] 0.19593782 0.12305126 0.12305126 0.12305126
0.05396993 [6] 0.07355409 0.07355409 0.07355409
> bd.psi.result$sigma
        [,1       [,2]
[1,] 0.06056659 0.02235376
[2,] 0.02235376 0.02164093
> bd.psi.result$psi
, , 1
        [,1]
[1,] 0.2465655
, , 2
        [,1]
[1,] 0.001661140
# likelihood values at each iteration:
> bd.psi.result$logll
[1] 53.76033 59.97957 65.15120 66.95766 67.42337
67.52435 [7] 67.54583 67.55096 67.55242 67.55289
67.55306 67.55313 [13]67.55315 67.55316
```

Regardless of the assumption on Ψ, exactly same estimates are obtained for β. The standard errors, however, differ. Differences are not large enough to make any substantive difference in concluding the significance of the treatment effects. The final model to be fit is the fixed-effects-only model:

```
# Model 3: fixed-effects only:
# below the argument maxits indicate the maximum
number of iterations and
```

```
# eps indicates the convergence criteria
no.psi.result<mlm.em(y,subj,pred,xcol,maxits=200,
eps=0.0001)
# results:
# estimates of the fixed-effects:
> no.psi.result$beta
          [,1]       [,2]
[1,] 1.80250000  5.8797145
[2,] -0.46625000 0.0748543
[3,]  0.06625000 0.1209458
[4,]  0.09970017 0.0899250
# standard errors of vec(beta):
> sqrt(diag(no.psi.result$xtwxinv))
[1]  0.19500522  0.27577902  0.27577902  0.27577902
0.05413598 [6] 0.07655984 0.07655984 0.07655984
> no.psi.result$sigma
          [,1]       [,2]
[1,] 0.30421628 0.04111835
[2,] 0.04111835 0.02344564
> no.psi.result$logoll
[1]  45.34702  45.18980  45.08828  45.05638  45.04649
45.04320  [7]  45.04203  45.04160  45.04144  45.04137
45.04135
```

While the similar substantive conclusions are seen, the standard errors of the $\mathrm{vec}(\hat{\beta})$ is considerably higher than the previous two estimates. Another interesting change is seen in the direction of the estimate of β_{41}. Treatment 2 effect still remains to be the only significant effect.

The likelihood-ratio (LR) statistic for testing the block-diagonal model against the unstructured alternative is 2(69.23369-67.55316)=3.36106; comparing this value to χ_1^2 yields a p-value of 0.0667. Inclusion of a nonzero Ψ_{12} would not make the model fit significantly better than the one with $\Psi_{12} = 0$ at $\alpha = 0.05$ level test. Conducting the test of H_0: $\Psi = 0$ require more careful handling as explained by Stram and Lee.[17] In such testing the difference in the -2logL(θ) under the null and alternative hypothesis is compared to the sampling distribution of the LR statistic under H_0. This is a 50:50 mixture of χ_3^2 and χ_0^2 for unstructured (there are three unknown parameters in Ψ. Testing unstructured Ψ versus fixed-effects-only model yields $-2\log(\mathrm{L}(\theta_{\Psi\neq0}) - L(\theta_{\Psi=0}) = 48.38468$, which yields a p-value of essentially zero. Similar result is obtained under a block-diagonal version of Ψ as well, indicating that in this application, block-diagonal version of Ψ is well suited under the realized data.

Functions *mlmmm.em*, *mlmmmbd.em and mlm.em* also provide users to see the missingness patterns by calling the attribute names rmat, and patt to see which missingness pattern each row of *y* belongs:

```
> unst.psi.result$rmat
   [,1] [,2]
[1,] 1   1
[2,] 0   1
[3,] 1   0
> unst.psi.result$patt
[1] 1 1 2 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 [28]1
1 1 2 1
> y[3,]


adg   initwt
NA   6.095825
> y[4,]
   adg   initwt
1.860000 5.913503
> y[5,]
 adg initwt
1.31    NA
```

EXAMPLE 2: LONGITUDINAL DATA APPLICATION

Data example used here come from the observational study on pediatric pain as reported by Fanurik, Zeltzer, Roberts and Blount.[18] The data consist of up to four observations on 64 children 8 to 10. The first response variable (*y.tot* below) is the length of time in seconds that the child can tolerate keeping his or her arm in very cold water, a proxy measure of pain tolerance. After cold becomes intolerable, the child removes his or her arm. The arm is toweled off and no harm is caused. Missing data exist due to absenteeism, broken arms, or other reasons. Two measures were taken during a first visit followed by two more measures during a second visit after a two-week gap. The second response variable is the pain tolerance on log scale (y.rat below).

During the first visit, children were classified into one of two groups: attenders and distracters according to the style of coping with the pain. The children were asked what they were thinking about during the trials. Those who were thinking about the experiment, the experimental apparatus, the feelings from their arms and so on were classified as attenders. Those who thought about other things were classified as distracters. A treatment was administered prior to fourth occasion, and the treatment consisted of a ten-minute counseling intervention to either attend, distract or no advice. The no advice treatment consisted of a discussion without advice regarding to any coping strategy. Interest lies in the main effects of treatment, style of coping and interactions between treatment and counseling, which was expected.

Another substantive interest is to jointly model pain rating and pain tolerance (on the log scale). Figure 4 displays this relationship over time for some children, showing the existence of child-specific effects. For example, while child 2, 4, 6 show a positive correlation of pain rating and tolerance over time, this is rather neutral for child 3 and 7, and negative for child 1 and 8. Fitting models with random slope allowing different growth trajectory for kids as well as random intercept model could be of interest.

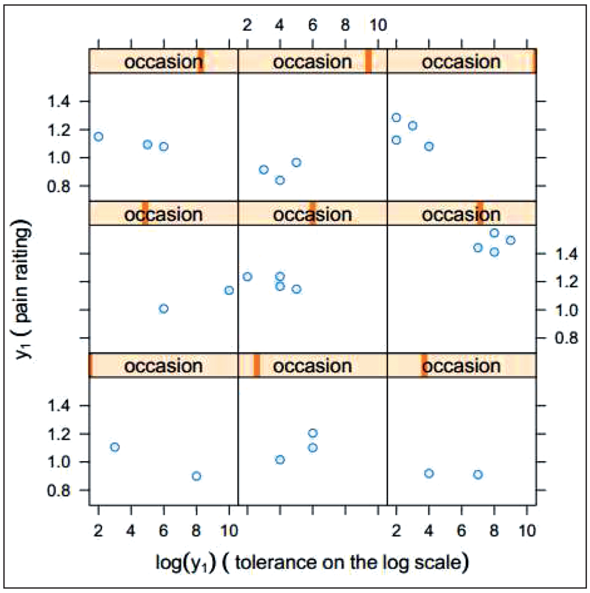

FIGURE 4: Relationship of y_1 and y_2 over four occasion for selected children.

SET-UP

Original data was in a short form, repeated observations were provided as a single entry, first step was to transform this data into the long form. The example data within **mlmmm** package is in the long form with up to four repeated observations for each child id. This data is called via

```
data(pain)
:
> data(pain)
> head(pain)
  y.tol y.rat intercept agey cs tmt.1 tmt.2 subj
1 3.022861   3     1     1 0    1     0    1
2 3.564166   4     1     1 0    1     0    1
3 2.659560   7     1     1 0    1     0    1
4 2.460443   8     1     1 0    1     0    1
5 3.336837   6     1     1 1    0     1    2
6 3.187179   5     1     1 1    0     1    2
# univariate summaries of the response variables:
> summary(pain$y.tol)
Min.  1st Qu.  Median   Mean  3rd Qu.   Max.  NA's
1.831  2.781   3.166  3.315  3.691   5.481 16.000
> summary(pain$y.rat)
Min.  1st Qu.  Median  Mean  3rd Qu.    Max.   NA's
1.000  4.000   5.500  5.544  7.000   10.000  15.000
```

Now form the response matrix *y*, design matrix *pred* and subject vector *subj*. I also create a "time" vector to be included in the pred matrix so that measurement times can be incorporated in fitting the models.

```
y=as.matrix(cbind(pain$y.tol,pain$y.rat))
```

```
# Note that tolerance is already on the log scale
#
# now form the pred matrix:
time=rep(c(1:4),64)
pred=as.matrix(cbind(pain$intercept,time,pain$agey,
pain$cs,pain$tmt.1,pain$tmt.2))
#
subj=pain$subj
xcol=1:ncol(pred)
zcol=1
```

MODEL FITTING

First, let's focus on the random-intercept model. Three models assuming unstructured, block-diagonal Ψ and fixed-effects models can be fit executing the following commands at the R prompt:

```
long.res=mlmmm.em(y,subj,pred,xcol,zcol,maxits=200,
eps=0.0001)
longbd.res=mlmmmbd.em(y,subj,pred,xcol,zcol,max-
its=200, eps=0.0001)
longfixed.res=mlm.em(y,subj,pred,xcol,maxits=200,
eps=0.0001)
#
> long.res$beta
        [,1]       [,2]
[1,] 2.78010388  5.84824480
[2,] -0.02358553 0.04208358
[3,] 0.31310707 -0.36155428
[4,] 0.40946967  0.10651299
[5,] 0.18166287 -0.47664034
[6,] 0.35006801 -0.15425094

> longbd.res$beta
        [,1]       [,2]
[1,] 2.78191892  5.83114245
[2,] -0.02403048 0.04440265
[3,] 0.30645353 -0.34965681
[4,] 0.40401472  0.10901264
[5,] 0.18845836 -0.47448911
[6,] 0.35568336 -0.14704982
```

```
> longfixed.res$beta
          [,1]       [,2]
[1,] 2.7835930  5.78420103
[2,] -0.0249679 0.04240086
[3,] 0.3091541 -0.30784237
[4,] 0.4314645  0.06151034
[5,] 0.1822180 -0.45719037
[6,] 0.3221639 -0.07203939
> sqrt(diag(long.res$xtwxinv))
[1] 0.22291995 0.02383897 0.17190954 0.16453747
0.19596008 [6] 0.19225994 0.58827661 0.09208623
0.44245824 0.42486540 [11]0.50663055 0.49779217
> sqrt(diag(longbd.res$xtwxinv))
[1] 0.22157118 0.02388268 0.17079559 0.16349014
0.19491453 [6] 0.19123704 0.58478712 0.09227789
0.43936606 0.42195669 [11]0.50366088 0.49489902
> sqrt(diag(longfixed.res$xtwxinv))
[1] 0.15978309 0.04112023 0.10048185 0.09641079
0.11513351 [6] 0.11353507 0.47047000 0.12107561
0.29586172 0.28387475 [11]0.33900248 0.33429598
```

Initial examination of the fixed-effects reveal that significant effects include the coping style (slight significance) on the pain tolerance. Age, treatment 2 appear to be slightly significant with the pain tolerance. The realized data fail to show significant association of time and treatment 1 with pain tolerance, as well as any association of time, age, coping style and treatment with pain rating. While pain tolerance is decreasing over time, coping pain is improving over time. Although not significant, pain rating is negatively impacted by age and treatment; positively impacted by time and coping style, as expected intuitively.

LR test can be performed using the last values of the output similar to the clustered data application given in Section 3:

```
> long.res$logll[long.res$iter]
[1] -254.1050
> longbd.res$logll[longbd.res$iter]
[1] -255.1398
> longfixed.res$logoll[longfixed.res$iter]
[1] -344.798
```

showing that the unstructured version of Ψ is not supported by data, in other words, the random-effects can be assumed independent apriori. Note that, under any of these three models, the error terms' covariance matrices reveal the negative correlation of residual terms of pain tolerance and pain coping:

```
> long.res$sigma
           [,1]       [,2]
[1,] 0.17190333 -0.09755071
[2,] -0.09755071 2.58807873
> longbd.res$sigma
          [,1]      [,2]
[1,] 0.1726225 -0.1057802
[2,] -0.1057802 2.5994805
> longfixed.res$sigma
       [,1       [,2]
[1,] 0.5224774 -0.2720314
[2,] -0.2720314 4.5297034
```

To see how the models with random intercept and random slope fit, we modify the argument *zcol*:

```
zcol=1:2
long.res=mlmmm.em(y,subj,pred,xcol,zcol,maxits=200, eps=0.0001)
longbd.res=mlmmmbd.em(y,subj,pred,xcol,zcol,maxits=200, eps=0.0001)
longfixed.res=mlm.em(y,subj,pred,xcol,maxits=200, eps=0.0001)
> long.res$logll[long.res$iter]
[1] -247.6662
> longbd.res$logll[longbd.res$iter]
[1] -249.6028
> longfixed.res$logoll[longfixed.res$iter]
[1] -344.798
```

Examining these log-likelihood values against the ones obtained under random-intercept only model show that the model allowing different intercepts as well as growth over time for children for pain tolerance and pain coping is better supported by data.

DISCUSSION

R package **mlmmm** presents an important computational tool to help researchers in the analysis of multivariate longitudinal or clustered data with missing values. It should be noted that this package can also be used as a great companion to packages such as **pan**, which is ideal for conducting inference via multiple imputation.[10] **mlmmm** should be used in incomplete-data problems consisting of many small clusters in which case its performance is near optimal. If the problem contains large clusters (e.g. multi-stage surveys) where large amounts of missing data are seen, MCMC-type algorithms should be preferred as the computational cost required by **mlmmm** can be high.

It is straightforward to show that the multivariate mixed-effects model underlying **mlmmm** implies a set conditional univariate models for each response variable given the others, where the others are incorporated into the columns of X_i. Thus, the estimation procedures implemented in **mlmmm** are also appropriate for longitudinal analyses with partially missing covariates, when those covariates are later going to be incorporated into an analytic model as linear fixed effects. Future extensions of **mlmmm** will address nonlinear associations and interactions, which is not possible to address in the current settings.

Future implementations will also address higher levels of clustering where missing values are not restricted to higher level of observational units (i.e. missing covariates at the subject or cluster level, for example, non-time-varying covariates). Finally note that models in **mlmmm** assume that the rows of y_i are conditionally independent given B with common covariance matrix $\sum$. In the univariate case, this assumption is commonly relaxed by allowing a residual covariance matrix of the form $\sigma^{|2}V_i$, where V_i has a simple (e.g., autoregressive or banded) pattern with a small number of unknown parameters. Sensible multivariate extensions of these patterned covariance structures produces models and algorithms that are complicated even apart from missing data. For example, the obvious extension of $vec(\epsilon_i) \sim N(0, (\sum \otimes I_{n_i}))$ to

$vec(\epsilon_i) \sim N(0, (\sum\otimes V_i))$ seems too restrictive for many longitudinal datasets, because the response variables $Y_1,\ldots,Y_r$, would be required to have an identical autocorrelations. Accounting for autocorrelated residuals in a plausible manner may prove be a daunting task in the multivariate case. In many cases, apparent nonzero correlations among the rows of $\in_i$ may arise because of a misspecified model for the mean structure over time. The problem may sometimes be reduced or eliminated by including additional (e.g., higher-order polynomial) terms for time in the covariate matrices X_i or Z_i.

Acknowledgement

This research was in part supported by the National Institute on Minority Health and Health Disparities, National Institutes of Health (grant number 1 P20 MD003373). The content is solely the responsibility of the authors and does not represent the official views of the National Institute on Minority Health and Health Disparities or the National Institutes of Health. I would like to thank Jing Hua Zhao for his help on packaging mlmmm. I would also like to thank Robert Weiss for providing the data used in Section 4.

REFERENCES

1. Dempster A, Laird N, Rubin D. Maximum likelihood from ıncomplete data via the EM algorithm. J Roy Statist Soc Ser B 1977;39(1): 38-1.
2. Rubin DB. Multiple Imputation for Nonresponse in Surveys. 1st ed. New York: John Wiley and Sons; 1987.
3. Rubin D. Multiple imputation after 18+ years. J Amer Statist Assoc 1996;91(434):473-89.
4. Little R, Rubin D. Statistical analysis with missing data. 1sted. New York: John Wiley & Sons; 1987. p.1-278.
5. Schafer J. Analysis of incomplete multivariate data. 1st ed. London: Chapman & Hall; 1997. p.1-448.
6. Schimert J, Schafer J, Hesterberg T, Fraley C, Clarkson D. Analyzing data with missing values in S-Plus. Seattle WA: Data Analysis Products Division, Insightful Corporation; 2000.
7. Raghunathan TE, Lepkowski JM, VanHoewyk J. A multivariate technique for multiply imputing missing values using a sequence of regression models. Survey Methodology 2001; 27(1):85-95.
8. Buuren SV, Groothuis-Oudshoorn K. mice: Multivariate imputation by chained equations in R. J Stat Softw 2011;45(3):1-67.
9. SAS Institute. SAS/Stat User's Guide, Version 8.2. 1st ed. Carey, NC: SAS Institute Incorporated; 2001. p.1-324.
10. Schafer J, Yucel R. Computational strategies for multivariate linear mixed-effects models with missing values. J Comput Graph Stat 2002;11(2):437-57.
11. Liu M, Taylor J, Belin T. Multiple imputation and posterior simulation for multivariate missing data in longitudinal studies. Biometrics 2000;56(4):1157-63.
12. Carpenter JR, Goldstein H, Kenward MG. REALCOM-IMPUTE Software for Multilevel Multiple Imputation with Mixed Response Types. J Stat Softw 2011;45(5):1-14.
13. Laird N, Ware J. Random-effects Models for Longitudinal Data. Biometrics 1982;38(4):963-74.
14. Jennrich R, Schluchter M. Unbalanced repeated-measures models with structured covariance matrices. Biometrics 1986;42: x805-20.
15. Shah A, Laird N, Schoenfeld D. A random-effects model for multiple characteristics with possibly missing data. J Am Stat Assoc 1997; 92(438):775-9.
16. Littell RC, Miliken GA, Stroup WW, Russell DW. SAS System for Mixed Models. 1sted. Carey, NC: SAS Institute, Incorporated; 1996. p.1-633.
17. Stram DO, Lee JW. Correction to "Variance Components Testing in the Longitudinal Mixed Effects Model". Biometrics 1995;51: 1196.
18. Fanurik D, Zeltzer L, Roberts M, Blount R. The relationship between childrens coping styles and psychological interventions for cold pressor pain. Pain 1993;53(2):213-22.