STAT 571 HW7

Wenxiao Gu

March 5, 2014

Contents

1 Relationship between fixed effects, random effects, GLS, and penalized regression; confounding; model misspecication

1.1 Part (a)

Figure 1 shows the marginal relationship between literacy and age regardless of individual. We connect the points of each individual.

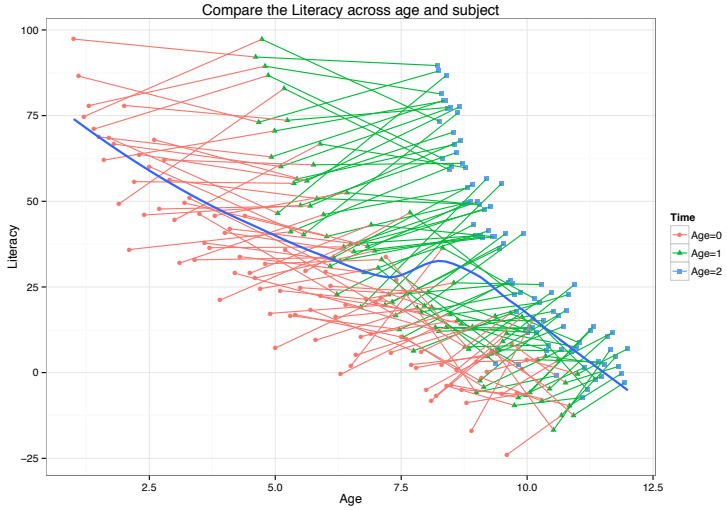

Figure 1 – The relationship between bias and the σ^2. The blue lowess curve shows the marginal trend of literacy v.s. age.

Figure 2 shows the time correlation within each individual.

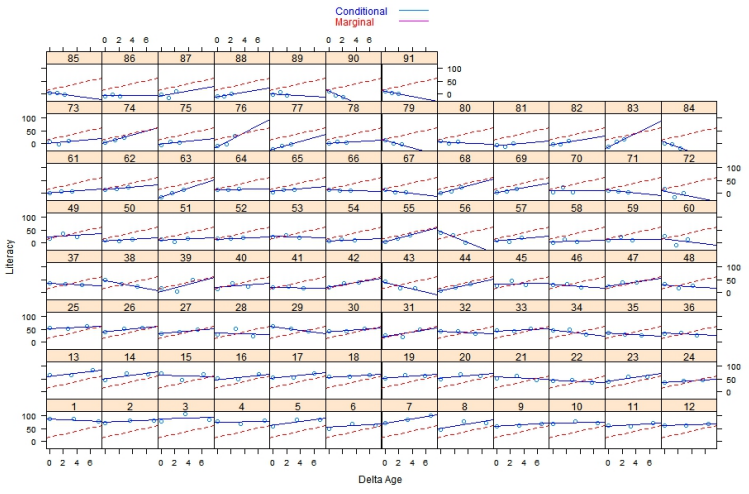

Figure 2 – Scatterplot for each individual. The within individual correlation over time is obvious.

1.2 Part (b and c)

We will compare the following two models: the fixed effect model (subject-specific model, i.e. there is an intercept β_i for each i),

$$\begin{aligned} Y_{ij} &= (\beta_i) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij}. \\ \epsilon_{ij} &\overset{iid}{\sim} \text{some distribution with } (0,\ \sigma^2) \end{aligned}$$

and the linear mixed model

$$\begin{aligned} Y_{ij} &= (\beta_0 + \gamma_i) \quad + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij} \\ \gamma_i &\overset{iid}{\sim} \quad N(0,\ \sigma_\gamma^2) \\ \epsilon_{ij} &\overset{iid}{\sim} \quad N(0,\ \sigma^2) \end{aligned}$$

The subject-specific model is better. Random effect model assumes that the random intercept is i.i.d. from some distribution. However, in this case since the data of X is fixed, the intercept for each subject is fixed, so random intercept model should not be used.

We simulated the data by the generating mechanism 5000 times. In each simulation, we fitted both models and recorded the estimates of β_L. The linear mixed model was done with REML.

1.3 Part (d)

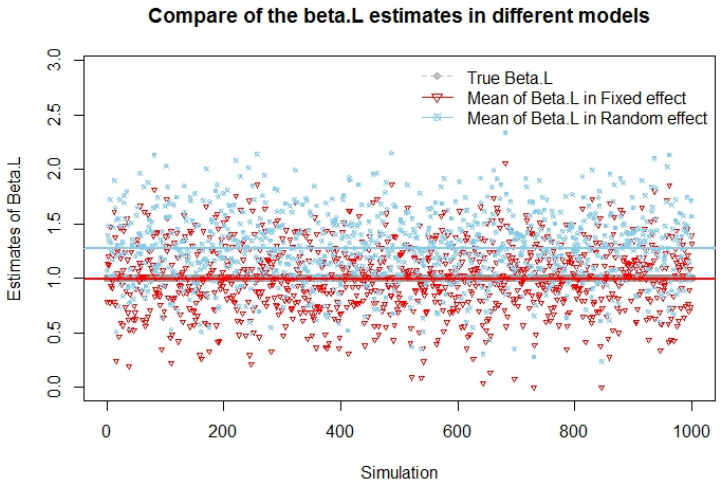

Figure 3 – The relationship between bias and the σ^2

In Figure 3 we see that the mean of $\hat{\beta}_L$ in the fixed effect model(subject-specific model) is closer to the true β_L.

Table 1 shows the Bias and standard error of of $\hat{\beta}_L$ in both models. The $\hat{\beta}_L$ in the fixed effect model seems to be unbiased, while the $\hat{\beta}_L$ in the random effect model does not.

Model	Bias	Standard deviation
subject-specific model	-1.27E-03	0.317
linear mixed model	2.65E-01	0.322

Table 1 – Bias and standard error of $\hat{\beta}_L$.

1.4 Part (e)

Table 2 shows the variables in the model. Both of the estimates of σ^2 are unbiased.

Model	Expected variance σ^2	Expected variance σ_γ^2
subject-specific model	99.94	-
linear mixed model	100.34	594.54

Table 2 – Estimated variance .

1.5 Part (f)

Calculate the "true" variance of $f(\cdot)$ function, given the "data" of the population.

$$\begin{aligned} Y_{ij} &= (\beta_0 + \gamma_i) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij} \\ EY_{ij} &= (\beta_0 + \gamma_i) + \beta_L \left(x_{i1} - x_{ij}\right) \\ \mu_\gamma &= E\left[f(x_{i1})\right] \\ \sigma_\gamma^2 &= var\left[f(x_{i1})\right] = \frac{1}{91}\left[\Sigma_i^{91}\left(f(x_{i1}) - \mu_\gamma\right)\right]^2 \\ &= 596.97 \end{aligned}$$

1.6 Part (g)

There are two ways to do. One is using the GLS regression and the other is penalized regression. From the below random effect model as asked, the only thing changed is the *exact* bias of $\hat{\beta}_L$,

$$Y_{ij} = (\beta_0 + \gamma_i) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij}.$$

- GLS

$$\begin{aligned} \hat{\beta}_L &= \left(X^T W X\right)^{-1}\left(X^T W Y\right) \\ W_i &= V_i^{-1} \\ var(Y_{ij}) &= cov\left(\gamma_i + \epsilon_{ij\cdot}, \gamma_i + \epsilon_{ij}\right) = \sigma^2 + \sigma_\gamma^2 \\ cov(Y_{ij}, Y_{ij'}) &= cov\left(\gamma_i + \epsilon_{ij'}, \gamma_i + \epsilon_{ij'}\right) = \sigma_\gamma^2 \\ \Sigma_i &= \begin{bmatrix} \sigma^2+\sigma_\gamma^2 & \sigma_\gamma^2 & \sigma_\gamma^2 \\ \sigma_\gamma^2 & \sigma^2+\sigma_\gamma^2 & \sigma_\gamma^2 \\ \sigma_\gamma^2 & \sigma_\gamma^2 & \sigma^2+\sigma_\gamma^2 \end{bmatrix} \\ &= \left(\sigma^2+\sigma_\gamma^2\right)\begin{bmatrix} 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 \end{bmatrix} \\ V_i &= \begin{bmatrix} 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 \end{bmatrix} \end{aligned}$$

The exact bias is

$$\begin{aligned} E(\hat{\beta}_L) - \beta_L &= \left(X^T W X\right)^{-1} \left(X^T W E(Y)\right) - \beta_L \\ &= \left(\Sigma_i X_i^T W_i X_i\right)^{-1} \left(\Sigma_i X_i^T W_i E\left(Y_i\right)\right) - \beta_L, \\ &= 1.27288 - 1 \\ &= 0.27288 \end{aligned}$$

where

$$E(Y) - \beta_L = (\beta_0 + \gamma_i) + \beta_L \left(x_{i1} - x_{ij}\right).$$

- Penalized regression
  From the lecture notes, we have the solution from the penalized regression,

$$\begin{aligned} \begin{bmatrix} \beta \\ \gamma \end{bmatrix} &= \arg\min \Sigma_i^n \left\{ \left(Y_i - X_i\beta - 1\gamma_i\right)^T \left(Y_i - X_i\beta - 1\gamma_i\right) + \frac{\sigma^2}{\sigma_\gamma^2}\gamma_i^2 \right\} \\ E\begin{bmatrix} \beta \\ \gamma \end{bmatrix} &= \arg\min \Sigma_i^n \left\{ \left(EY_i - X_i\beta - 1\gamma_i\right)^T \left(EY_i - X_i\beta - 1\gamma_i\right) + \frac{\sigma^2}{\sigma_\gamma^2}\gamma_i^2 \right\} \end{aligned}$$

  - As in GLS case, we plug the mean of Y into the penalized package. The output is 0.27288.

1.7 Part (h)

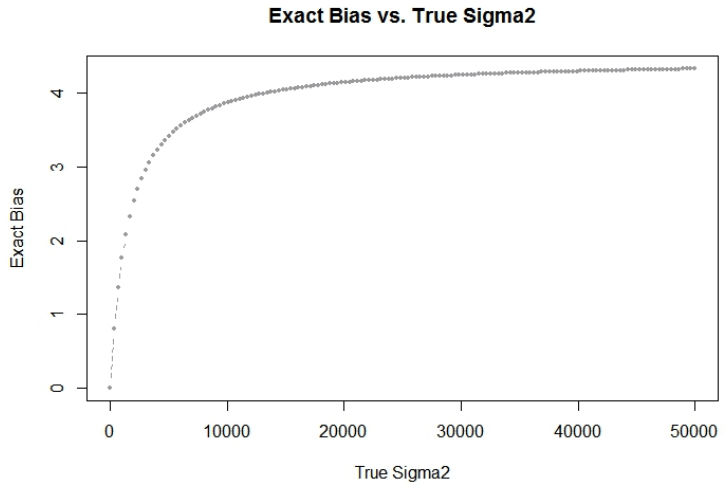

Figure 4 – The relationship between bias and the σ^2

Figure 4 shows that the exact bias is increasing as the σ^2 increases. But the slope is first decreasing.

1.8 Part (i)

$$
\begin{aligned}
Y_{ij} &= (\beta_i) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij} && (1)\\
Y_{ij} &= (\beta_0 + \gamma_i) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij} && (2)\\
Y_{ij} &= (\beta_0) + \beta_L \left(x_{i1} - x_{ij}\right) + \epsilon_{ij} && (3)\\
E\begin{bmatrix} \beta \\ \gamma \end{bmatrix} &= \arg\min \Sigma_i^n \left\{ \left(EY_i - X_i\beta - 1\gamma_i\right)^T \left(EY_i - X_i\beta - 1\gamma_i\right) + \frac{\sigma^2}{\sigma_\gamma^2}\gamma_i^2 \right\} && (4)\\
V_i &= \begin{bmatrix} 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} \\ \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & \frac{\sigma_\gamma^2}{\sigma^2+\sigma_\gamma^2} & 1 \end{bmatrix} && (5)
\end{aligned}
$$

Table 3 shows the comparison of the GLS and the penalized regression in explaining for the patterns in Figure 4. When $\sigma^2 = 0$, the estimates of β is unbiased and thus the bias is zero. If σ^2 increases from zero, the bias increases and converges to some value.

Case	Explanation
$\sigma^2 \approx 0$	$\frac{\sigma^2}{\sigma_\gamma^2}$ is too small and Eq. (4) is a unconstraint problem for a subject-specific fixed effect model of Eq. (1).
$\sigma^2 \approx 0$	In GLS, Eq. (5) would make the all the within-cluster observations highly dependent, resulting in a high weight on
	within-cluster contrast. This is the subject-specific fixed effect model of Eq. (1).
$\sigma^2 \approx +\infty$	$\frac{\sigma^2}{\sigma_\gamma^2}$ is too large and Eq. (4) would make the $\gamma_i = 0$. This is the OLS model of Eq. (3).
$\sigma^2 \approx +\infty$	In GLS, Eq. (5) would make the all the within-cluster observations independent. This is the OLS model of Eq. (3).

Table 3 – Summary of the GLS and the penalized regression

1.9 Part (j)

The GLS gives unbiased estimate of β with any working covariance matrix when mean is correctly specified. In this case, we fit the mixed model (wrong mean model) and estimateβ using GLS; we fit the subject-specific intercept fixed model (correct mean model) using OLS. So the mixed model with working independence would not give same mean of estimate (in fact it is more biased); the subject-specific intercept fixed model would give the same mean of estimate even if we used GLS to estimate β.

2 Fitting and interpreting the results of a linear mixed effects model; deriving the REML and ML likelihood functions; robust standard error estimation

2.1 Part (a)

Table 4 shows the variables in the model.

Variable name	Explanation
scr	The response
kidney	Kidney = 1 means the subject with hereditary disease.
age	Age of the subject
id	Identification number for subject

Table 4 – Summary of the variables in the model

We denote jth ($j = 1, ..., m_i$) *observation* of the ith ($i = 1, ..., 619$) *subject (id)* as Y_{ij}. We fit the following model of *scr* on *age*, *kidney disease status*, and their interaction. Since we are interested to estimate the rate of change in serum creatinine *scr* for *subjects* with and without hereditary kidney disease, then the random effects are on *age*. We include an random slope and an random intercept to account for correlation within *subject*. The variable *kidney* is a indicator, and the value equal to *1* means that the subject has kidney disease. In our data, the kidney disease status did not change within subject.

$$\begin{aligned} Y_{ij} &= \beta_0 + \beta_1 \cdot age_{ij} + \beta_2 \cdot kidney_i + \beta_{12} \cdot age_{ij} \times kidney_i \\ &\quad + \gamma_{i0} + \gamma_{i1} \cdot age_{ij} + W_i(t_{ij}) + \epsilon_{ij} \\ \begin{pmatrix} \gamma_{i0} \\ \gamma_{i1} \end{pmatrix} &\overset{iid}{\sim} N\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, G = \begin{pmatrix} G_{11} = \sigma^2_{\gamma_0} & 0 \\ 0 & G_{22} = \sigma^2_{\gamma_1} \end{pmatrix} \right] \end{aligned}$$

We include serial dependence W_i to construct the covariance of ϵ_i, which is a multivariate Gaussian with zero mean and covariance as below,

$$\begin{aligned} E(\epsilon_{ij}) &= 0 \\ var(\epsilon_{ij}) &= \sigma^2 \\ cov(\epsilon_{ij}, \epsilon_{ij'}) &= \begin{cases} (1 - nugget) \cdot \sigma^2 \left[1 - 1.5 \cdot \frac{|age_{ij} - age_{ij'}|}{range} + 0.5 \left(\frac{|age_{ij} - age_{ij'}|}{range} \right)^3 \right] & |age_{ij} - age_{ij'}| < range \\ 0 & |age_{ij} - age_{ij'}| \geqslant range \end{cases} \end{aligned}$$

where σ^2 is the residual variance, and the *nugget* parameter that measures the portion of σ^2 that belongs to the nugget effect.

Moreover, the matrix notation $\epsilon_i \sim N(0, R_i)$ where R_i is the covariance matrix of ϵ_i given as below,

$$\begin{aligned} R_{i,jj'} &= cov(\epsilon_{ij}, \epsilon_{ij'}) \\ R_{i,jj} &= var(\epsilon_{ij}). \end{aligned}$$

2.2 Part (b)

Within a subject i, let X_i be the design matrix of fixed effect covariates, and Z_i be the design matrix of random effect covariates, as below

$$
\begin{aligned}
X_i &= \left\{ \begin{array}{cccc} 1 & age_{i1} & kidney_i & age_{i1} \times kidney_i \\ 1 & age_{i2} & kidney_i & age_{i2} \times kidney_i \\ \vdots & \vdots & \vdots & \vdots \\ 1 & age_{im} & kidney_i & age_{im} \times kidney_{im} \end{array} \right\} \\
Z_i &= \left\{ \begin{array}{cc} 1 & age_{i1} \\ 1 & age_{i2} \\ \vdots & \vdots \\ 1 & age_{im} \end{array} \right\}.
\end{aligned}
$$

Let $\beta = (\beta_0, \beta_1, \beta_2, \beta_3)^T$ and $\gamma_i = (\gamma_{i0}, \gamma_{i1})^T$. Our data generating model is below,

$$
\begin{aligned}
Y_i &= X_i\beta + Z_i\gamma_i + \epsilon_i \\
\gamma_i &\sim N(0, G) \\
\epsilon_i &\sim N(0, R_i)
\end{aligned}
$$

Or equivalently if we set $\alpha = \left(\sigma^2, nugget, range, \sigma^2_{\gamma_0}, \sigma^2_{\gamma_1}\right)^T$, we have

$$
\begin{aligned}
Y_i &= X_i\beta + Z_i\gamma_i + \epsilon_i \\
cov(\gamma_i) &= G(\alpha) \\
cov(\epsilon_i) &= R_i(\alpha) \\
cov(Y_i) &= Z_iG(\alpha)Z_i^T + R_i(\alpha) = \Sigma_i(\alpha)
\end{aligned}
$$

The ML log likelihood function from profiling likelihood (3.48) is below,

$$
\begin{aligned}
l(\alpha) &= -\Sigma_i^n log\,|\Sigma_i| - \Sigma_i^n \left(Y_i - X_i\hat{\beta}\right)^T \Sigma_i^{-1} \left(Y_i - X_i\hat{\beta}\right) \\
\hat{\beta} &= \beta(\alpha) = \left(\Sigma_i^n X_i^T\Sigma_i^{-1}X_i\right)^{-1} \left(\Sigma_i^n X_i^T\Sigma_i^{-1}Y_i\right).
\end{aligned} \tag{6}
$$

The REML log likelihood is below,

$$
\begin{aligned}
l(\alpha) &= -\Sigma_i^n log\,|\Sigma_i| - \Sigma_i^n log\left|X_i^T\Sigma_i^{-1}X_i\right| - \Sigma_i^n \left(Y_i - X_i\hat{\beta}\right)^T \Sigma_i^{-1} \left(Y_i - X_i\hat{\beta}\right) \\
\hat{\beta} &= \beta(\alpha) = \left(\Sigma_i^n X_i^T\Sigma_i^{-1}X_i\right)^{-1} \left(\Sigma_i^n X_i^T\Sigma_i^{-1}Y_i\right).
\end{aligned} \tag{7}
$$

To obtain ML and REML $\hat{\alpha}$'s, we maximize Eq. (6) and Eq. (7) respectively.

2.3 Part (c)

The results are shown in the below Table Table 5. The estimates of $\alpha = \left(\sigma^2, nugget, range, \sigma^2_{\gamma_0}, \sigma^2_{\gamma_1}\right)^T$ via REML are slightly higher than the ones using MLE.

Table 5 – Estimated variance parameters for ML and REML

Linear mixed-effects model fit by **maximum likelihood**
Data: data
Log-likelihood: 71.38007
Fixed: scr ~ age * kidney
(Intercept) age kidney age:kidney
1.196490051 -0.003275613 0.314170354 -0.016642243
Random effects:
Formula: ~1 + age | id
Structure: Diagonal
(Intercept) age Residual
StdDev: 0.1553242 0.004798277 0.2115859 (Sqaure)
Correlation Structure: Spherical spatial correlation
Formula: ~age | id
Parameter estimate(s):
range nugget
7.3676230 **0.27**63843
Number of Observations: 1585
Number of Groups: 619
> fit.ml$sigma^2
[1] 0.0448

Linear mixed-effects model fit by **REML**
Data: data
Log-restricted-likelihood: 53.66149
Fixed: scr ~ age * kidney
(Intercept) age kidney age:kidney
1.19663113 -0.00327953 0.31431681 -0.01664654
Random effects:
Formula: ~1 + age | id
Structure: Diagonal
(Intercept) age Residual
StdDev: 0.1559501 0.004816316 0.2117685 (Sqaure)
Correlation Structure: Spherical spatial correlation
Formula: ~age | id
Parameter estimate(s):
range nugget
7.3776035 **0.27**58132
Number of Observations: 1585
Number of Groups: 619
> fit.reml$sigma^2
[1] 0.0448

	$\hat{\sigma}^2$	$\sigma^2_{\gamma_0}$	$\sigma^2_{\gamma_1}$	$\hat{nugget}$	$\hat{range}$
REML	0.0448	0.0243	2.320e-05	0.276	7.38
ML	0.0448	0.0241	2.302e-05	0.276	7.37

2.4 Part (d)

Table 6 shows the values of three versions of standard estimates for ML and REML. Values in Table 6a are generally smaller than in Table 6b.

ML

	Estimates	SE:model-based	SE:empirical sand	SE:model-based sand
(intercept)	1.1965	5.3871E-02	5.0702E-02	5.3871E-02
age	-3.2756E-03	1.4232E-03	1.2062E-03	1.4232E-03
kidney	3.1417E-01	7.0909E-02	6.8759E-02	7.0909E-02
age:kidney	-1.6642E-02	1.8420E-03	1.7155E-03	1.8420E-03

(a) ML

REML

	Estimates	SE:model-based	SE:empirical sand	SE:model-based sand
(intercept)	1.1966	5.3998E-02	5.0725E-02	5.3998E-02
age	-3.2795E-03	1.4267E-03	1.2068E-03	1.4267E-03
kidney	3.1432E-01	7.1077E-02	6.8775E-02	7.1077E-02
age:kidney	-1.6647E-02	1.8466E-03	1.7159E-03	1.8466E-03

(b) REML

Table 6 – Summary of three versions of standard estimates for ML and REML

The REML standard error estimates are larger than the ML, because the estimated variance parameters are also slightly larger as we have seen in Part (C).

Model-based sandwich standard errors and model-based standard errors in each of the tables are the same.

If we use "summary(fit, adjustSigma=T)", then columns 2 and 4 in ML are not EXACTLY the same, but they are close because of a degree-of-freedom correction.

2.5 Part (e)

Table 7 gives point estimates and robust sandwich standard error estimates that account for clustering for the marginal rates of change in *scr* in subjects with and without kidney disease for the model fit by REML. From the derivation below, we have construted the combination to get the rate of change in *scr* in subjects with and without kidney disease, as well as the robust sandwich standard errors of the rate of change.

$$\begin{aligned} \hat{\beta}_{age}(Desease = T) &= \hat{\beta}_1 + \hat{\beta}_{12} \\ &= -0.01993 \end{aligned}$$

$$\begin{aligned} Var\left(\hat{\beta}_{age}(Desease = T)\right) &= Var_{robust-sand}(\hat{\beta}_1 + \hat{\beta}_{12}) \\ &= (0,1,0,1) \times Var(\hat{\beta}) \times (0,1,0,1)^T \\ &= (1.2197E-03)^2 \end{aligned}$$

	REML	
	Estimates	SE: robust/empirical sand
Without kidney disease	-3.2795E-03	1.2068E-03
With kidney disease	-1.9926E-02	1.2197E-03

Table 7 – Point estimates and robust sandwich standard error estimates from REML

2.6 Part (f)

The subject 20 in the data has kidney disease at three different ages that are not 60 years old. The model should include age and kidney disease status and subject id. To predict scr for a similar subject with kidney disease at age 60, we need not only the fixed effect coeffcient estimates but the estimated values of the random intercept and slope of subject 20.

Table 8 shows both estimates of fixed and random effects for the subject at age 60.

	Fixed (subj 20)	Predicted value (random person similar)
With kidney disease	0.3154	0.9682

Table 8 – Subject 20 and a similar subject at age 60.

Appendix

```
## Problem 1
library(nlme)
library(Matrix)
library(penalized)
fx = function (x){
  out = (10-x)^2
  return(out)
}
# n.subject = 91
# m.time = 3
genXY = function(n.subject= 91,m.time= 3,sigma2=100,beta.L=1){
  x = matrix(0,n.subject,m.time)
  for (i in 1:n.subject){
    for (j in 1:m.time){
      if ( j == 1){
        # baseage
        x[i,j] = 1+0.1*(i-1)
        delta.xi = (1+ (10-x[i,1])/10)^2
      } else{
        # j=2, j=3
        x[i,j] = x[i,1] + (j-1) * delta.xi
      }
    }
  }
  xx=cbind(x,x-x[,1])
  colnames(xx) = c("base.age","age2","age3","base.age-base.age","delta.age","2delta.age")
  xx
#   beta.L = 1
#   sigma2 = 100
  sigma = sqrt(sigma2)
#
  y = y.m = subject.long =  time.long =rep(0,n.subject * m.time)
  base.age.long = delta.age.long =rep(0,n.subject * m.time)
  for (i in 1:n.subject){
    for (j in 1:m.time){
     y.m[j + (i-1)*m.time] = fx(x[i,1]) + beta.L * (x[i,j]-x[i,1])

     y[j + (i-1)*m.time] = rnorm(1,0,sigma) +  y.m[j + (i-1)*m.time]
     subject.long[j + (i-1)*m.time] = i
     time.long[j + (i-1)*m.time] = j
     base.age.long[j + (i-1)*m.time] = xx[i,1]
     delta.age.long[j + (i-1)*m.time] = xx[i,3+j]
    }
  }
  df.long = data.frame(subject = subject.long,
                       time=time.long,base.age = base.age.long,
                       delta.age =delta.age.long, literacy = y,
                       y.m = y.m)
  return(list(df.long = df.long,x=x,xx=xx,y=y,y.m=y.m))
}
xx=genXY()$xx
y.m = genXY()$y.m
```

```
# sim.out = genXY()
n.sim = 1500
sim.out = list()
bl.lm.0 = bl.lm.fix = bl.lme.ran = rep(0,n.sim)
sig2.lm.fix = sig2.lme.ran = rep(0,n.sim)
sig2.subj.ran = rep(0,n.sim)
for (i in 1:n.sim){
  sim.out[[i]] = genXY()
  tmr = sim.out[[i]]
  df.long = tmr$df.long

  lm.fit = lm(literacy~ delta.age,data=df.long)
  bl.lm.0[i]= lm.fit$coef[2]

  # fit fixed effects linear reg
  lm.fit.fix = lm(literacy~ delta.age+as.factor(subject),data=df.long)
  bl.lm.fix[i]= lm.fit.fix$coef[2]
  sig2.lm.fix[i]= summary(lm.fit.fix)$sigma^2

  # fit random effects linear reg
  lme.fit.ran = lme(literacy~delta.age,
                    random=~1|as.factor(subject),
                    data=df.long)
  bl.lme.ran[i]=fixef(lme.fit.ran)["delta.age"]
  sig2.subj.ran[i]=getVarCov(lme.fit.ran)[[1]]
  sig2.lme.ran[i]= summary(lme.fit.ran)$sigma^2


}

library(lattice)
library(latticeExtra)
a1=xyplot(literacy ~ delta.age | factor(subject),
          data=df.long,
          main=NULL,
          ylab="Literacy",xlab="Delta Age",
          panel = function(x,y) {
            j = panel.number()
            panel.xyplot(x,y)
            panel.lmline(x,y,col="blue") # fit scatter per grp
            panel.abline(lm(literacy ~ delta.age,data=df.long),lty=2,col="Red") # ignore grp
          },
          auto.key=list(text=c("Conditional","Marginal"),
                        lines=T,points=F, space = "top",col=c("blue","red")
          ))
print(a1)

df = data.frame(x=rep(xx[,1],3),y=c(y1,y2,y3),group=rep(1:3,each=length(xx[,1])))
m=ggplot(df)
m=m+
  geom_point(aes(x=x,y=y,group=factor(group),colour=factor(group),shape=factor(group)),size=4,alpha = 0.
#   stat_binhex(aes(x=x,y=y,group=factor(group),colour=factor(group),shape=factor(group)))
  theme_bw()+xlab("Baseline Age")+ylab("F(xi1)")+
```

```
  ggtitle("Compare the F(xi1) across true, fixed effect and random effect")+
  #   guides(colour=guide_legend(title="Time"))+
  #   guides(shape=guide_legend(title="Time"))+
  #   geom_smooth(aes(x=age,y=literacy,group=factor(time),shape=factor(time)),se=FALSE,size=1)
#   geom_line(aes(x=age,y=literacy,group=factor(subject),colour=factor(time),shape=factor(time)))+
  scale_colour_discrete(name="Model",
                        breaks=1:3,
                        labels=c("True","fixed effect","random effect"))+
  scale_shape_discrete(name="Model",
                       breaks=1:3,
                       labels=c("True","fixed effect","random effect"))
m2 = m + geom_smooth(aes(x=x,y=y),data=subset(df,group==1),se=TRUE,size=1,method="auto")
m2

# lme.fit.ran$coef
# tmr = fixef(lme.fit.ran)[1]+ranef(lme.fit.ran)[,1]
# points(x,tmr,pch=22)
sim.table = cbind(1:length(bl.lm.0),bl.lm.0,bl.lm.fix,sig2.lm.fix,bl.lme.ran,sig2.lme.ran,sig2.subj.ran)
# table for ggplot2
apply(sim.table,2,mean)
apply(sim.table,2,sd)
sim.table2 = as.data.frame(cbind(rep(1:length(bl.lm.0),3),
                                 c(rep(1,length(bl.lm.fix)),bl.lm.fix, bl.lme.ran),
                                 rep(0:2,each=length(bl.lm.0))))
colnames(sim.table2) = c("simulation","beta.L","model")
?as.data.frame

sim.table=as.data.frame(sim.table)
head(sim.table)
apply(sim.table,2,mean)
# bl.lm.0     bl.lm.fix   sig2.lm.fix    bl.lme.ran  sig2.lme.ran sig2.subj.ran
# 750.5000000     5.4584807     0.9888249    99.9400113     1.2642132   100.3440212   594.5482125

apply(sim.table,2,sd)
# bl.lm.0     bl.lm.fix   sig2.lm.fix    bl.lme.ran  sig2.lme.ran sig2.subj.ran
# 433.1570154     0.2905246     0.3175003    10.4622072     0.3218348    10.5467856    33.8165254

tail(sim.table2)
plot(1:n.sim,rep(1,n.sim),
     xlab = "Simulation",
     ylab = "Estimates of Beta.L",
     ylim = c(min(bl.lm.fix,bl.lme.ran),max(bl.lm.fix,bl.lme.ran)*1.25),
     pch = 19,"b",lty = 2, lwd= 2, col = grey(0.6),
     main="Compare of the beta.L estimates in different models")
points(1:n.sim,bl.lm.fix,pch = 6,lty = 1, col = "red", cex = 0.5)

points(1:n.sim,bl.lme.ran,pch = 13,lty = 1, col = "skyblue",cex = 0.5)

abline(h=mean(bl.lm.fix),lty = 1, col = "red",lwd=2)

abline(h=mean(bl.lme.ran),lty = 1, col = "skyblue",lwd=2)
legend("topright",
       legend =c("True Beta.L","Mean of Beta.L in Fixed effect","Mean of Beta.L in Random effect"),
```

```
       lty = c(2,1,1),
       col = c("grey","red","skyblue"),
       pch= c(19,6,13),
       bty="n")




# m=ggplot(sim.table2)
# m=m+geom_point(aes(x=simulation,y=beta.L,group=factor(model),colour=factor(model),shape=factor(model))
#   theme_bw()+xlab("Simulation")+ylab("Estimates of Beta.L")+
#   ggtitle("Compare of the beta.L estimates in different models")+
#   guides(colour=guide_legend(title="Model"))+
#   guides(shape=guide_legend(title="Model"))+
#   geom_smooth(aes(x=simulation,y=beta.L,group=factor(model),colour=factor(model)),se=FALSE,size=1)
# m

df.long$age = df.long$base.age+df.long$delta.age
col = c("red","blue","yellow")
tmr = df.long[which(df.long$subject==1),]
m=ggplot(df.long)
m=m+geom_point(aes(x=age,y=literacy,group=factor(time),colour=factor(time),shape=factor(time)))+
  theme_bw()+xlab("Age")+ylab("Literacy")+
  ggtitle("Compare the Literacy across age and subject")+
#   guides(colour=guide_legend(title="Time"))+
#   guides(shape=guide_legend(title="Time"))+
#   geom_smooth(aes(x=age,y=literacy,group=factor(time),shape=factor(time)),se=FALSE,size=1)
  geom_line(aes(x=age,y=literacy,group=factor(subject),colour=factor(time),shape=factor(time)))+
 scale_colour_discrete(name="Time",
                       breaks=1:3,
                       labels=c("Age=0","Age=1","Age=2"))+
  scale_shape_discrete(name  ="Time",
                       breaks=c(1:3),
                       labels=c("Age=0","Age=1","Age=2"))
m2 = m + geom_smooth(aes(x=age,y=literacy),se=FALSE,size=1)
m2
ggsave(m,file="literacy.pdf")
ggsave(m2,file="literacy2.pdf")
# warnings()




exact.bias = function (sigma2,df.long,beta.L){
  ## exact
  true.sigma2 = sigma2
  true.sigma2.gamma = var(fx(xx[,1]))*90/91
  # 596.9742
# find the exact bias using GLS
  X = cbind(1, df.long[,4])
  Y = df.long$y.m
  Sigma = matrix(true.sigma2.gamma,nrow=3,ncol=3)
  diag(Sigma) = true.sigma2.gamma+true.sigma2
```

```
  W = solve(Sigma)
  Sigma.all = matrix(0,91*3,91*3)
  for (i in 1:n.subject){
    Sigma.all[3*(i-1)+(1:3),3*(i-1)+(1:3)] = W
  }
  true.beta.1=(solve(t(X)%*%Sigma.all%*%X)%*%t(X)%*%Sigma.all%*%Y)[2]
  print(true.beta.1)
  #1.273

  # find the exact bias by penalized OLS
  pen.fit = penalized(df.long$y.m,
                      penalized=~as.factor(df.long$subject),
                      unpenalized=~df.long$delta.age,
                      lambda2=true.sigma2 /true.sigma2.gamma)
  true.beta.1.pen = coef(pen.fit)[2]
  print(true.beta.1.pen)
  bias=true.beta.1-beta.L
  return(bias)
}

beta.L = 1
n.sigma2 = 150
sigma2.lst = bias.lst = seq(1,50000,l=n.sigma2)

for (i in 1:n.sigma2 ){
  sigma2 = sigma2.lst [i]
  df.long=genXY(sigma2=sigma2,beta.L=beta.L)$df.long
  bias.lst[i] = exact.bias(sigma2=sigma2,df.long=df.long,beta.L=beta.L)
}

plot(sigma2.lst,bias.lst,
     xlab = "True Sigma2",
     ylab = "Exact Bias",
#      ylim = c(min(bl.lm.fix,bl.lme.ran),max(bl.lm.fix,bl.lme.ran)*1.25),
     pch = 19,"b",lty = 2, lwd= 1, col = grey(0.6),cex=0.6,
     main="Exact Bias vs. True Sigma2")

## Problem 2
library(nlme)

data = read.csv("creat.csv")
head(data)
data$group = as.numeric(data$group)
data$kidney[data$group<3]=1
data$kidney[data$group>2]=0
data$interaction = data$kidney*data$age

fit.ml=lme(fixed = scr~age*kidney,
    random = reStruct(~1+age| id, pdClass="pdDiag"),
    corr=corSpher(form = ~age | id, nugget=TRUE),
    method = "ML",
    data=data
    )
```

```
# fit.ml$sigma^2
fit.reml=lme(fixed = scr~age*kidney,
    random = reStruct(~1+age| id, pdClass="pdDiag"),
    corr=corSpher(form = ~age | id, nugget=TRUE),
    method = "REML",
    data=data
)
# fit.reml$sigma^2

## 2d)
# > head(data)
# id group    age   scr kidney interaction
# 1  1     1 35.765 0.182      1      35.765
# 2  1     1 37.990 0.088      1      37.990
# 3  2     2 24.997 1.429      1      24.997
# 4  2     2 27.441 1.111      1      27.441
# 5  2     2 30.524 1.429      1      30.524
# 6  3     1 51.083 0.156      1      51.083

ses = function (fit,data){
  A = matrix(0,4,4)
  B = matrix(0,4,4)
  B.mb = matrix(0,4,4)
  #get variance components
  #range
  range = as.numeric(coef(fit$modelStruct, unconstrained=F)[3])
  #nugget effect
  nugget = as.numeric(coef(fit$modelStruct, unconstrained=F)[4])
  # beta est.
  beta = fixef(fit)
  # how many subj.
  id.lst = unique(data$id)
  n = length(id.lst)
  # get var(gamma)
  G = matrix(c(getVarCov(fit)),2,2)
  #residual variance
  sigma2 = fit$sigma^2

  # loop in the subjects
  for(i in 1:n){
    yi = data$scr[which(data$id==id.lst[i])]
    mi = length(yi)
    head(data)
    tmr = data[which(data$id==id.lst[i]),c(3,5,6)]
    xi = cbind(rep(1,mi),tmr)
    # design mat
    zi = model.matrix(~1+xi$age)
    Ri = matrix(0,mi,mi)

    ## calculate rho matrix : loop within subject
    for(j in 1:mi){
      for(k in 1:mi){
        # eps over age
```

```
        if(abs(xi$age[j]-xi$age[k]) < range){
          rho = (1-nugget)*( 1 - 1.5*abs(xi$age[j]-xi$age[k])/range +
                                 0.5*abs(xi$age[j]-xi$age[k])^3/range^3
                           )
        }else{
          rho = 0
        }
        if(j == k){
          rho = 1
        }
        Ri[j, k] = sigma2*rho
      }
    }
    if(mi==1){
      si = zi%*%G%*%t(zi) + Ri # Simga_i matrix in the hw
    }else{
      si = zi%*%G%*%t(zi) + Ri
    }

    xi = as.matrix(xi)
    b = as.matrix(beta,1,4)
    resi = yi - xi%*%b
    B = B + t(xi)%*%solve(si)%*%resi%*%t(resi)%*%solve(si)%*%xi
    B.mb = B.mb + t(xi)%*%solve(si)%*%xi # canceled out.
    A = A + t(xi)%*%solve(si)%*%xi
  }
  se.mb =  sqrt(diag(summary(fit)$varFix))
  sand = solve(A)%*%B%*%solve(A)
  sand.mb = solve(A)%*%B.mb%*%solve(A)
  output = matrix(0,4,4)
  summary(fit)
  output[,1] = beta
  output[,2] = se.mb
  output[,3] = sqrt(diag(sand))
  output[,4] = sqrt(diag(sand.mb))
  colnames( output ) = c("Estimate","SE:model-based","SE:empirical sand","SE:model-based sand")
  rownames( output ) = c("(intercept)","age","kidney","age:kidney")
  output
  return(list(output=output,sand=sand))
}

T1=ses(fit.ml,data)$output # table of ML
# Estimate SE:model-based SE:empirical sand SE:model-based sand
# (intercept)  1.196490051    0.053870837       0.050701519         0.053870837
# age         -0.003275613    0.001423158       0.001206204         0.001423158
# kidney       0.314170354    0.070908899       0.068759444         0.070908899
# age:kidney  -0.016642243    0.001841956       0.001715466         0.001841956
T2=ses(fit.reml,data)$output# table of REML
# Estimate SE:model-based SE:empirical sand SE:model-based sand
# (intercept)  1.19663113    0.053997862       0.050725204         0.053997862
# age         -0.00327953    0.001426745       0.001206847         0.001426745
# kidney       0.31431681    0.071077369       0.068774636         0.071077369
# age:kidney  -0.01664654    0.001846595       0.001715854         0.001846595
```

```
reml.sand = ses(fit.reml,data)$sand
reml.sand

## 2e)
# without kidney disease
T2[,c(1,3)]
# Estimate SE:empirical sand
# (intercept)  1.19663113       0.050725204
# age         -0.00327953       0.001206847
# kidney       0.31431681       0.068774636
# age:kidney  -0.01664654       0.001715854

# with kidney disease
sqrt(c(0,1,0,1)%*%reml.sand%*%c(0,1,0,1)) #0.001219702
fixef(fit.ml)[2]+fixef(fit.ml)[4] #-0.01991786

## 2f)
data[which(data$id==20),]
# id group    age   scr kidney interaction
# 42 20     1 30.034 1.429      1      30.034
# 43 20     1 44.397 1.429      1      44.397
# 44 20     1 49.884 1.111      1      49.884

fixef(fit.ml) #fixed
# subject 20 at 60 years old
age.pred = 60
kidney.pred = 1
inter.pred = age.pred*kidney

fixed.20 = c(1,age.pred,kidney.pred,inter.pred)%*%as.numeric(fixef(fit.ml)) #0.3156
rownames(ranef(fit.ml))[20]
random.20 = unlist(ranef(fit.ml)[20,])%*%c(1,age.pred)
random.20
random.20 + fixed.20# 0.9676927
```