Mixed Model Analysis of Censored Longitudinal Data with Flexible Random Effects Density - Supplemental Material

David M. Vock, Marie Davidian, Anastasios A. Tsiatis,
North Carolina State University, Raleigh, NC 27695, USA
Andrew J. Muir
Duke Clinical Research Institute, Durham, NC 27705, USA

1 Technical Details for Asymptotic Bias

Let $Y_{k,o}$, $Y_{k,c}$, $Q_{k,o}$, $Q_{k,c}$, $l_{k,o}$, $l_{k,c}$, and Σ be as defined in the manuscript. Define $V_{k,o}$ to be the design matrix corresponding to the non-censored observations under censoring pattern k., If we let $\Sigma_{(k,c),(k,c)}$ to be the rows and columns of Σ corresponding to $q_{k,c}$ and similarly define $\Sigma_{(k,o),(k,o)}$ and $\Sigma_{(k,c),(k,o)}$, then we can define $\Sigma^*_{(k,c),(k,c)} = \Sigma_{(k,c),(k,c)} - \Sigma_{(k,c),(k,o)}\Sigma^{-1}_{(k,o),(k,o)}\Sigma_{(k,o),(k,c)}$, $V^*_{k,c} = V_{k,c} - \Sigma_{(k,c),(k,o)}\Sigma^{-1}_{(k,o),(k,o)}V_{k,o}$, and $Q^*_{k,c} = Q_{k,c} - \Sigma_{(k,c),(k,o)}\Sigma^{-1}_{(k,o),(k,o)}Q_{k,o}$.

If we set the expectation of the partial derivative with respect to δ of the log likelihood equal to zero, we obtain

$$\begin{aligned} 0 &= \mathrm{E}_{Q|V}\left[I(Q > l)\frac{\partial}{\partial\delta}\left\{\log f_Y(Q;\delta,V)\right\}\right] & (1)\\ &+ \sum_{k=1}^{2^n-2}\mathrm{E}_{Q|V}\left[I(Q_{k,c} = l_{k,c}, Q_{k,o} > l_{k,o})\frac{\partial}{\partial\delta}\left\{\log f_{Y_{k,o}}(Q_{k,o};\delta,V_{k,o})\right\}\right] \\ &+ \sum_{k=1}^{2^n-2}\mathrm{E}_{Q|V}\left[I(Q_{k,c} = l_{k,c}, Q_{k,o} > l_{k,o})\frac{\partial}{\partial\delta}\left\{\log F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)\right\}\right] \\ &+ \mathrm{E}_{Q|V}\left[I(Q = l)\frac{\partial}{\partial\delta}\left\{\log F_Y(l;\delta,V)\right\}\right\}. \end{aligned}$$

The first term in (1) can be re-written as

$$\begin{aligned} & \mathrm{E}_{Q|V}\left[I(Q > l)\frac{\partial}{\partial\delta}\left\{\log f_Y(Q;\delta,V)\right\}\right] & (2)\\ =& \int_l^\infty V^T\Sigma^{-1}(q - V\delta)g_Y(q,\delta^*,V)dq \\ =& P(Q > l)V^T\Sigma^{-1}(V\delta^* - V\delta) + \int_l^\infty V^T\Sigma^{-1}(q - V\delta^*)g_Y(q,\delta^*,V)dq, \end{aligned}$$

where $P(\cdot)$ is the probability of the given event under the true random effects density.

Similarly if $f_Y(q;\delta,V)$ is the multivariate normal density, the last term of (1) can be written as

$$\begin{aligned} & \mathrm{E}_{Q|V}\left[I(Q = l)\frac{\partial}{\partial\delta}\left\{\log F_Y(l;\delta,V)\right\}\right\} & (3)\\ =& G_Y(l;\delta^*,V)\frac{\frac{\partial}{\partial\delta}F_Y(l;\delta,V)}{F_Y(l;\delta,V)} \\ =& \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta,V)}\frac{\partial}{\partial\delta}\left\{\int_{-\infty}^l f_Y(q;\delta,V)dq\right\} \end{aligned}$$

$$
\begin{aligned}
&= \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta,V)} \int_{-\infty}^{l} \frac{\partial}{\partial\delta}\left\{f_Y(q;\delta,V)\right\} dq \\
&= \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta,V)} \int_{-\infty}^{l} \frac{\partial}{\partial\delta}\left\{\log f_Y(q;\delta,V)\right\} f_Y(q;\delta,V) dq \\
&= \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta,V)} \int_{-\infty}^{l} V^T\Sigma^{-1}(q-V\delta) f_Y(q;\delta,V) dq
\end{aligned}
$$

Let $c_0(\delta;q,V) = \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta,V)} V^T\Sigma^{-1}(q-V\delta)f_Y(q;\delta,V)$. Then if we take the first-order Taylor expansion of $c_0(\delta;q,V)$ about $\delta=\delta^*$, then (3) can be written as

$$
\begin{aligned}
&\int_{-\infty}^{l} \frac{G(l;\delta^*,V)}{F_Y(l;\delta^*,V)} V^T\Sigma^{-1}(q-V\delta^*) f_Y(q;\delta^*,V) dq + \int_{-\infty}^{l} \frac{\partial}{\partial\delta}\left\{c_0(\delta;q,V)\right\}\Big|_{\delta=\delta_0} (\delta-\delta^*) dq \qquad (4)\\
&= \int_{-\infty}^{l} V^T\Sigma^{-1}(q-V\delta^*) \left\{ \frac{G_Y(l;\delta^*,V)}{F_Y(l;\delta^*,V)} f_Y(q;\delta^*,V) - g_Y(q;\delta^*,V)\right\} dq \\
&\qquad + \int_{-\infty}^{l} \frac{\partial}{\partial\delta}\left\{c_0(\delta;q,V)\right\}\Big|_{\delta=\delta_0} (\delta-\delta^*) dq + \int_{-\infty}^{l} V^T\Sigma^{-1}(q-V\delta^*) g_Y(q,\delta^*,V) dq,
\end{aligned}
$$

for some δ_0 interior to the line segment joining δ and δ^*.

The summand in the third term of equation (1) can be simplified as follows:

$$
\begin{aligned}
& \mathrm{E}_{Q|V}\left[I(Q_{k,c}=l_{k,c}, Q_{k,o}>l_{k,o}) \frac{\partial}{\partial\delta}\left\{\log F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)\right\}\right] \qquad (5)\\
&= \mathrm{E}_{Q|V}\left\{ I(Q_{k,c}=l_{k,c}, Q_{k,o}>l_{k,o}) \frac{\frac{\partial}{\partial\delta} F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}\right\} \\
&= \mathrm{E}_{Q_{k,o}|V}\mathrm{E}_{Q_{k,c}|Q_{k,o},V}\left\{ I(Q_{k,c}=l_{k,c}, Q_{k,o}>l_{k,o}) \frac{\frac{\partial}{\partial\delta} F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}\right\} \\
&= \mathrm{E}_{Q_{k,o}|V}\left\{ I(Q_{k,o}>l_{k,o}) G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta^*,V) \frac{\frac{\partial}{\partial\delta} F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|Q_{k,o};\delta,V)}\right\} \\
&= \int_{l_{k,o}}^{\infty} \left\{ G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V) \frac{\frac{\partial}{\partial\delta} F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta,V)}\right\} g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o}) dq_{k,o} \\
&= \int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}} \frac{G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta,V)} \frac{\partial}{\partial\delta}\left\{\log f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V)\right\} \times \\
&\qquad\qquad f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V) dq_{k,c} g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o}) dq_{k,o}
\end{aligned}
$$

We can also write the summand in the second term of (1) as

$$
\begin{aligned}
& \mathrm{E}_{Q|V}\left[I(Q_{k,c}=l_{k,c}, Q_{k,o}>l_{k,o}) \frac{\partial}{\partial\delta} \log\left\{f_{Y_{k,o}}(Q_{k,o},\delta,V_{k,o})\right\}\right] \qquad (6)\\
&= \int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}} \frac{\partial}{\partial\delta}\log\left\{f_{Y_{k,o}}(Q_{k,o},\delta,V_{k,o})\right\} g_Y(q;\delta^*,V) dq
\end{aligned}
$$

Therefore, if we subtract

$$
\begin{aligned}
& \int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}} \frac{\partial}{\partial\delta}\left\{\log f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V)\right\} g_Y(q;\delta^*,V) dq \qquad (7)\\
&= \int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}} \frac{\partial}{\partial\delta}\left\{\log f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V)\right\} g_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V) dq_{k,c} g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o}) dq_{k,o}
\end{aligned}
$$

from (5) and add the same quantity to (6) then we obtain the following.

The summand for the second term with adding (7) is now

$$\begin{aligned}
&\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}\frac{\partial}{\partial\delta}\left[\log\left\{f_{Y_{k,o}}(q_{k,o},\delta,V_{k,o})\right\}+\log\left\{f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V)\right\}\right]g_Y(q;\delta^*,V)dq \qquad (8)\\
=\ &\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}\frac{\partial}{\partial\delta}\left\{\log f_Y(q,\delta,V)\right\}g_Y(q;\delta^*,V)dq\\
=\ &\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}V^T\Sigma^{-1}(q-V\delta)g_Y(q;\delta^*,V)dq\\
=\ &V^T\Sigma^{-1}(V\delta^*-V\delta)P(Q_{k,c}=l_{k,c},Q_{k,o}>l_{k,o})+\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}V^T\Sigma^{-1}(q-V\delta^*)g_Y(q;\delta^*,V)dq.
\end{aligned}$$

The summand for the third term (with subtracting 7) is

$$\begin{aligned}
&\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}V_{k,c}^{*T}\Sigma^{*-1}_{(k,c),(k,c)}\{q_{k,c}^*-V_{k,c}^*\delta\}\times \qquad (9)\\
&\left\{\frac{G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta,V)}f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta,V)-g_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V)\right\}dq_{k,c}g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o})dq_{k,o}
\end{aligned}$$

If we let $c_k(\delta;q,V)$ be the integrand in (9) and then take the first-order Taylor expansion about $\delta=\delta^*$ of $c_k(\delta;q,V)$, then (9) can be written as

$$\begin{aligned}
&\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}V_{k,c}^{*T}\Sigma^{*-1}_{(k,c),(k,c)}(q_{k,c}^*-V_{k,c}^*\delta^*)\times \qquad (10)\\
&\left\{\frac{G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V)}f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V)-g_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V)\right\}dq_{k,c}g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o})dq_{k,o}\\
+\ &\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}\frac{\partial}{\partial\delta}\left\{c_k(\delta;q,V)\right\}\bigg|_{\delta=\delta_0}(\delta-\delta^*)dq.
\end{aligned}$$

There is a slight abuse of notation here since the d_0 here need not be the same one in (4) nor the same for each k.

Combining (2), (4), (8), and (10) and doing a small amount of simplification gives

$$\begin{aligned}
0\ =\ &\{1-G_Y(l;\delta^*,V)\}V^T\Sigma^{-1}V(\delta^*-\delta) \qquad (11)\\
-\ &\int_{-\infty}^{l}\frac{\partial}{\partial\delta}\left\{c_0(\delta;q,V)\right\}\bigg|_{\delta=\delta_0}(\delta^*-\delta)dq\\
-\ &\sum_{k=1}^{2^n-2}\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}\frac{\partial}{\partial\delta}\left\{c_k(\delta;q,V)\right\}\bigg|_{\delta=\delta_0}(\delta^*-\delta)dq\\
+\ &\int_{-\infty}^{l}V^T\Sigma^{-1}(q-V\delta^*)\left\{\frac{G_Y(l;\delta^*,V)}{F_Y(l,\delta^*,V)}f_Y(q;\delta^*,V)-g_Y(q;\delta^*,V)\right\}dq\\
+\ &\sum_{k=1}^{2^n-2}\left[\int_{l_{k,o}}^{\infty}\int_{-\infty}^{l_{k,c}}V_{k,c}^{*T}\Sigma_{k,c}^{*-1}(q_{k,c}^*-V_{k,c}^*\delta^*)\times\right.\\
&\left.\left\{\frac{G_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o};\delta^*,V)}{F_{Y_{k,c}|Y_{k,o}}(l_{k,c}|q_{k,o},\delta^*,V)}f_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V)-g_{Y_{k,c}|Y_{k,o}}(q_{k,c}|q_{k,o};\delta^*,V)\right\}dq_{k,c}g_{Y_{k,o}}(q_{k,o};\delta^*,V_{k,o})dq_{k,o}\right].
\end{aligned}$$

Thus, $D(\delta^*, V)$ is given by

$$\begin{aligned} D(\delta^*, V) &= \{1 - G_Y(l; \delta^*, V)\} V^T \Sigma^{-1} V \\ &- \int_{-\infty}^{l} \frac{\partial}{\partial \delta} \{c_0(\delta; q, V)\}\bigg|_{\delta=\delta_0} dq \\ &- \sum_{k=1}^{2^n-2} \int_{l_{k,o}}^{\infty} \int_{-\infty}^{l_{k,c}} \frac{\partial}{\partial \delta} \{c_k(\delta; q, V)\}\bigg|_{\delta=\delta_0} dq. \end{aligned}$$

2 SAS Code

Here we show how PROC NLMIXED can be used to to obtain the SNP estimates for $q = 1$ and $q = 2$ ($K = 1$ and $K = 2$ for each). For $q = 1$ we fit a random intercept model and $q = 2$ we fit a random slope and intercept model. Gaussian quadrature was used to approximate the integral in equation(4.1)in the manuscript where the quadrature points are centered on the empirical Bayes estimates from assuming Gaussian random effects and scaled using the square root of the estimated variance matrix of the empirical Bayes estimates.

In the following code we make two calls to NLMIXED. The first call is used to obtain the empirical Bayes estimates. The second implements SNP estimation.

Prior to running NLMIXED, three data sets must be created. Two of those data sets should contain the starting values for the two calls to NLMIXED while the other should contain the data. The data set, referred to as start_value1 in the code below, should contain the starting values for the call to NLMIXED to obtain the empirical Bayes estimates. The data set start_value2 should have starting values for the call to NLMIXED to obtain the SNP estimates. Rather than pre-specifying starting values for the second call the NLMIXED, we show below how one can derive starting values for μ and r using the parameter estimates from the first call to NLMIXED and a grid of starting values for ξ.

In general, data sets for starting values should have a different variable for each parameter. If there are multiple observations in this data set, NLMIXED evaluates the log likelihood at each one, and then uses the starting value set with the largest log likelihood for optimization. The parameters that will need starting values, in addition to any parameters associated with additional covariates in the model, are indicated above the example code for the model. The other data set, referred to as data_set in the code, should contain the following variables:

- cresponse = The response value if not censored and the limit of quantification if the response is censored.
- censored = An indicator variable if the response is censored.
- subject = Subject ID.
- last_obs = An indicator variable if the observation is the last in the data set for the corresponding subject.
- time = The random slope variable.
- Any additional covariates to be included in the mean model should also be included in this data set.

2.1 $q = 2$ Code

We provide code for the following subject-specific intercept and slope semiparametric model, which is a specific example of model (2.1) in the main text:

$$Y_{ij} = b_{1i} + b_{2i} t_{ij} + \epsilon_{ij}, \tag{12}$$

where Y_{ij} is the response of interest for subject i at the j^{th} time; t_{ij} is the time; x_{ij} is null; $s_{ij} = (1, t_{ij})^T$; $\epsilon_{ij} \overset{iid}{\sim} N(0, \sigma^2)$; and $b_i = (b_{1i}, b_{2i})^T$ is the vector of subject specific intercept and slope which we assume can be written as $b_i = \mu + RZ_i$ with $Z_i = (Z_{1i}, Z_{2i})^T$, $\mu = (\mu_1, \mu_2)^T$, and R a (2×2) lower triangular matrix. We assume that Z_i follows the SNP density for the K described below. We do not observe Y_{ij} but instead observe Q_{ij} which equals Y_{ij} if Y_{ij} is larger than the lower limit of quantification and equal to the lower limit of quantification otherwise. The code here is easily adapted for cases when x_{ij} contains additional covariates.

In the following SAS code, we use following notation for variable names and parameters.

Parameter or Variable	SAS Notation
Q_{ij}	cresponse
t_{ij}	time
E(b_{1i})	b1mean
E(b_{2i})	b2mean
var(b_{1i})	D11
var(b_{2i})	D22
cov(b_{1i}, b_{2i})	D12
$\mu^T = (\mu_1, \mu_2)$	(mu_1,mu_2)
$\xi^T = (\xi_1, \ldots, \xi_d)$	(xi_1,...,xi_d)
$r^T = (R_{11}, R_{21}, R_{22})$	(R_11,R_21,R_22)
$Z_i^T = (Z_{1i}, Z_{2i})$	(z_1,z_2)
σ^2	sigma_sq

2.1.1 Obtain Empirical Bayes Estimates

The following obtains the empirical Bayes estimates of b_i from assuming the density of b_i is Gaussian. Note that this code also gives the $K = 0$ estimates as well. For this optimization, start_value1 should contain parameter estimates for the mean of the subject-specific intercept (b1mean) and slope (b2mean), the variance of those random effects (D11 and D22, respectively), the correlation between those random effects (Dcor), the variance of the intra-subject error (sigma_sq), as well as parameters for any other covariates included in the mean model.

```
proc nlmixed data=simdataset;
parms / data=start_value1;
bounds -1<Dcor<1, D11 D22 sigma_sq>=0;
pi=2*arsin(1);

/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=b1mean+b2mean*time+z_1+time*z_2;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
D12=(D11*D22)**0.5*Dcor;
model response ~ general(loglik);
random z_1 z_2 ~ normal([0,0], [D11,D12,D22]) subject=subject;
estimate "D12" (D11*D22)**0.5*Dcor;
predict z_1 out=int_data;
predict z_2 out=slope_data;
predict z_1+z_2 out=add_data;
ods output ParameterEstimates=parm_k0 AdditionalEstimates=ae_k0;
run;
```

The subsequent code is used to merge all the empirical Bayes estimates together into one data set, take the square root of the estimated variance matrix of the empirical Bayes estimate, and then combine with the data_set.

```
proc transpose data=parm_k0 out=tparm_k0;
var Estimate;
id Parameter;
run;

proc transpose data=ae_k0 out=tae_k0;
var Estimate;
id Label;
run;

data tparm_k0;
merge tparm_k0 tae_k0;
test=1;
run;

data int_data2;
set int_data (keep=subject Pred StdErrPred last_obs);
/* Only keep one observation per subject */
where last_obs=1;
rename Pred=blup_1;
blup_var1=StdErrPred**2;
drop StdErrPred;
run;

data slope_data2;
set slope_data (keep=subject Pred StdErrPred last_obs);
/* Only keep one observation per subject */
where last_obs=1;
rename Pred=blup_2;
blup_var2=StdErrPred**2;
drop StdErrPred;
run;

data add_data2;
set add_data (keep=subject Pred StdErrPred last_obs);
/* Only keep one observation per subject */
where last_obs=1;
rename Pred=blup_add;
blup_var_add=StdErrPred**2;
drop StdErrPred;
run;

data subjspec;
merge int_data2 slope_data2 add_data2;
```

```
by subject;
test = 1;
run;

data subjspec;
merge subjspec tparm_k0;
by test;
run;

data subjspec;
set subjspec;
/*Obtain the covariance between the empirical Bayes estimates */
blup_cov = (blup_var_add-blup_var1-blup_var2)/2;
drop last_obs _NAME_;
run;

data subjspec;
set subjspec;
lambda1 = ((blup_var1+blup_var2)+sqrt((blup_var2-blup_var1)**2+4*blup_cov**2))/2;
lambda2 = ((blup_var1+blup_var2)-sqrt((blup_var2-blup_var1)**2+4*blup_cov**2))/2;
Q11 = sqrt(1/(1+(blup_var1-lambda1)**2/blup_cov**2));
Q21 = -(blup_var1-lambda1)*Q11/blup_cov;
Q12 = sqrt(1/(1+(blup_var1-lambda2)**2/blup_cov**2));
Q22 = -(blup_var1-lambda2)*Q12/blup_cov;
/* Note: Gij is the (i,j)th element of the square root matrix */
G11 = sqrt(lambda1)*Q11**2+sqrt(lambda2)*Q12*Q21;
G12 = sqrt(lambda1)*Q11*Q12+sqrt(lambda2)*Q12*Q22;
G21 = sqrt(lambda1)*Q11*Q21+sqrt(lambda2)*Q21*Q22;
G22 = sqrt(lambda1)*Q12*Q21+sqrt(lambda2)*Q22**2;
beta_hat_1 = blup_1+b1mean;
beta_hat_2 = blup_2+b2mean;
run;

data subjspec2;
set subjspec (keep=subject G11 G12 G21 G22 beta_hat_1 beta_hat_2);
run;

data data_set2;
merge data_set subjspec2;
by subject;
run;
```

2.1.2 Obtain Grid of Starting Values

For the SNP estimation, start_value2 must contain starting values for μ, r, ξ, and σ^2. In many instances we may wish to consider a grid of values for ξ, use starting values for μ and r that give the same estimate of the mean and variance of the random effects as the estimates when $K = 0$, and then use the same starting value for σ^2 and any additional starting values for additional covariates as the estimates computed above. If the data set xi_matrix contains the grid of starting values for ξ and start_value1 contains just one set of starting values the following code can be used to obtain the corresponding estimates for μ and r.

Code for $K = 1$.

```
data xi_matrix;
set xi_matrix;
test=1;
run;

data start_value2;
merge tparm_k0 xi_matrix;
by test;
run;

data start_value2;
set start_value2;
a00=sin(xi_1);
a10=cos(xi_1)*sin(xi_2);
a01=cos(xi_1)*cos(xi_2);
z1mean = 2*a00*a10;
z2mean = 2*a00*a01;
z11var = a00**2+3*a10**2+a01**2;
z12var = 2*a10*a01;
z22var = a00**2+a10**2+3*a01**2;
A=(z11var-z1mean**2);
B=(z12var-z1mean*z2mean);
D=(z22var-z2mean**2);
R_11 = sqrt(D11/A);
B_q = 2*B*D12/(A*R_11)-2*D12*B/(A*R_11);
A_q = (B/A)**2-2*B**2/A+D;
C_q = A*(D12/(A*R_11))**2-D22;
R_22 = (-B_q+sqrt(B_q**2-4*A_q*C_q))/(2*A_q);
R_21 = (D12-B*R_11*R_22)/(A*R_11);
mu_1 = b1mean-R_11*z1mean;
mu_2 = b2mean-R_21*z1mean-R_22*z2mean;
run;

data start_value2;
/* Any other parameters for additional covariates should also be kept here*/
set start_value2 (keep=mu_1 mu_2 R_11 R_21 R_22 xi_1 xi_2 sigma_sq);
run;
```

Code for $K = 2$.

```
data xi_matrix;
set xi_matrix;
test=1;
run;

data start_value2;
merge tparm_k0 xi_matrix;
by test;
run;

data start_value2;
set start_value2;
c1=sin(xi_1);
```

```
c2=cos(xi_1)*sin(xi_2);
c3=cos(xi_1)*cos(xi_2)*sin(xi_3);
c4=cos(xi_1)*cos(xi_2)*cos(xi_3)*sin(xi_4);
c5=cos(xi_1)*cos(xi_2)*cos(xi_3)*cos(xi_4)*sin(xi_5);
c6=cos(xi_1)*cos(xi_2)*cos(xi_3)*cos(xi_4)*cos(xi_5);
a00=1.3683057*c1-0.25272473*c4-0.25272473*c6;
a10=c2;
a01=c3;
a20=-0.25272473*c1+0.65861913*c4-0.04848765*c6;
a11=c5;
a02=-0.25272473*c1+0.65861913*c6-0.04848765*c4;
z1mean = 2*a00*a10+2*a11*a01+2*a10*a02+6*a20*a10;
z2mean = 2*a00*a01+2*a10*a11+2*a01*a20+6*a01*a02;
z11var = a00**2+3*a10**2+a01**2+15*a20**2+3*a11**2+3*a02**2+6*a00*a20+6*a20*a02+2*a00*a02;
z12var = 2*a10*a01+2*a00*a11+6*a20*a11+6*a11*a02;
z22var = a00**2+a10**2+3*a01**2+3*a20**2+3*a11**2+15*a02**2+2*a00*a20+6*a00*a02+6*a20*a02;
A=(z11var-z1mean**2);
B=(z12var-z1mean*z2mean);
D=(z22var-z2mean**2);
R_11 = sqrt(D11/A);
B_q = 2*B*D12/(A*R_11)-2*D12*B/(A*R_11);
A_q = (B/A)**2-2*B**2/A+D;
C_q = A*(D12/(A*R_11))**2-D22;
R_22 = (-B_q+sqrt(B_q**2-4*A_q*C_q))/(2*A_q);
R_21 = (D12-B*R_11*R_22)/(A*R_11);
mu_1 = b1mean-R_11*z1mean;
mu_2 = b2mean-R_21*z1mean-R_22*z2mean;
run;

data start_value2;
/* Any other parameters for additional covariates should also be kept here*/
set start_value2 (keep=mu_1 mu_2 R_11 R_21 R_22 xi_1 xi_2 xi_3 xi_4 xi_5 sigma_sq);
run;
```

2.1.3 SNP Estimation

The following gives the code for the implementation of SNP. The code for $K = 1$ will be given first followed by $K = 2$.

```
proc nlmixed data=data_set2 noad;
parms / data=start_value2;
pi = 2*arsin(1);
bounds -1.5707<= xi_1 <= 1.5707,-1.5707 <= xi_2 <= 1.5707;
J = beta_hat_1-mu_1+G11*q_1+G12*q_2;
K = beta_hat_2-mu_2+G21*q_1+G22*q_2;
z_1 = J/R_11;
z_2 =-R_21/(R_11*R_22)*J+K/R_22;
a00=sin(xi_1);
a10=cos(xi_1)*sin(xi_2);
a01=cos(xi_1)*cos(xi_2);
p_k=(a00+a10*z_1+a01*z_2);
logden = log((p_k)**2);
```

```
re.dens = -0.5*(z_1**2+z_2**2);
q.norm = 0.5*(q_1**2+q_2**2);
log.det.iG = log((G11*G22-G12*G21));
log.det.R.inv = log(1/(R_11*R_22));

/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=mu_1+(R_11*z_1)+mu_2*time+(R_21*z_1+R_22*z_2)*time;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
if last_obs=1 then loglik=loglik+logden+re.dens+q.norm+log.det.iG+log.det.R.inv;
model cresponse ~ general(loglik);
random q_1 q_2 ~ normal([0,0],[1,0,1]) subject=subject;
z1mean = 2*a00*a10;
z2mean = 2*a00*a01;
z11var = a00**2+3*a10**2+a01**2;
z12var = 2*a10*a01;
z22var = a00**2+a10**2+3*a01**2;
A=(z11var-z1mean**2);
B=(z12var-z1mean*z2mean);
D=(z22var-z2mean**2);
estimate "b1mean" mu_1+R_11*z1mean;
estimate "b2mean" mu_2+R_21*z1mean+R_22*z2mean;
estimate "D11" R_11**2*A;
estimate "D12" A*R_11*R_21+B*R_11*R_22;
estimate "D22" A*R_21**2+2*B*R_21*R_22+D*R_22**2;
run;
```

SNP estimation for $K = 2$.

```
proc nlmixed data=data_set2 noad;
parms / data=start_value2;
pi = 2*arsin(1);
bounds -1.5707<= xi_1 <= 1.5707,-1.5707 <= xi_2 <= 1.5707, -1.5707 <= xi_3 <= 1.5707,
 -1.5707 <= xi_4 <= 1.5707, -1.5707 <= xi_5 <= 1.5707;
J = beta_hat_1-mu_1+G11*q_1+G12*q_2;
K = beta_hat_2-mu_2+G21*q_1+G22*q_2;
z_1 = J/R_11;
z_2 =-R_21/(R_11*R_22)*J+K/R_22;
c1=sin(xi_1);
c2=cos(xi_1)*sin(xi_2);
c3=cos(xi_1)*cos(xi_2)*sin(xi_3);
c4=cos(xi_1)*cos(xi_2)*cos(xi_3)*sin(xi_4);
c5=cos(xi_1)*cos(xi_2)*cos(xi_3)*cos(xi_4)*sin(xi_5);
c6=cos(xi_1)*cos(xi_2)*cos(xi_3)*cos(xi_4)*cos(xi_5);
a00=1.3683057*c1-0.25272473*c4-0.25272473*c6;
a10=c2;
a01=c3;
a20=-0.25272473*c1+0.65861913*c4-0.04848765*c6;
a11=c5;
a02=-0.25272473*c1+0.65861913*c6-0.04848765*c4;
```

```
p_k=(a00+a10*z_1+a01*z_2+a20*z_1**2+a11*z_1*z_2+a02*z_2**2);
logden = log((p_k)**2);
re.dens = -0.5*(z_1**2+z_2**2);
q.norm = 0.5*(q_1**2+q_2**2);
log.det.iG = log((G11*G22-G12*G21));
log.det.R.inv = log(1/(R_11*R_22));

/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=mu_1+(R_11*z_1)+mu_2*time+(R_21*z_1+R_22*z_2)*time;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
if last_obs=1 then loglik=loglik+logden+re.dens+q.norm+log.det.iG+log.det.R.inv;
model cresponse ~ general(loglik);
random q_1 q_2 ~ normal([0,0],[1,0,1]) subject=subject;
z1mean = 2*a00*a10+2*a11*a01+2*a10*a02+6*a20*a10;
z2mean = 2*a00*a01+2*a10*a11+2*a01*a20+6*a01*a02;
z11var = a00**2+3*a10**2+a01**2+15*a20**2+3*a11**2+3*a02**2+6*a00*a20+6*a20*a02+2*a00*a02;
z12var = 2*a10*a01+2*a00*a11+6*a20*a11+6*a11*a02;
z22var = a00**2+a10**2+3*a01**2+3*a20**2+3*a11**2+15*a02**2+2*a00*a20+6*a00*a02+6*a20*a02;
A=(z11var-z1mean**2);
B=(z12var-z1mean*z2mean);
D=(z22var-z2mean**2);
estimate "b1mean" mu_1+R_11*z1mean;
estimate "b2mean" mu_2+R_21*z1mean+R_22*z2mean;
estimate "D11" R_11**2*A;
estimate "D12" A*R_11*R_21+B*R_11*R_22;
estimate "D22" A*R_21**2+2*B*R_21*R_22+D*R_22**2;
run;
```

2.2 $q = 1$ Code

We provide code for the following random intercept semiparametric model, which is a specific example of model (2.1) in the main text:

$$Y_{ij} = b_{1i} + \beta_2 t_{ij} + \epsilon_{ij}, \tag{13}$$

where Y_{ij} is the response of interest for subject i at the j^{th} time; t_{ij} is the time; $x_{ij} = t_{ij}$; $s_{ij} = 1$; $\epsilon_{ij} \overset{iid}{\sim} N(0, \sigma^2)$; and $b_i = b_{1i}$ is the subject-specific intercept which we assume can be written as $b_i = \mu + RZ_i$. We assume that Z_i follows the SNP density for the K described below. We do not observe Y_{ij} but instead observe Q_{ij} which equals Y_{ij} if Y_{ij} is larger than the lower limit of quantification and equal to the lower limit of quantification otherwise. The code here is easily adapted for cases when x_{ij} contains additional covariates.

In the following SAS code, we use following notation for variable names and parameters.

2.2.1 Obtain Empirical Bayes Estimates

```
proc nlmixed data=data_set;
parms /data=start_value1;
bounds D11 sigma_sq>=0;
pi=2*arsin(1);
```

Parameter or Variable	SAS Notation
Q_{ij}	cresponse
t_{ij}	time
$\mathrm{E}(b_{1i})$	b1mean
β_2	b2mean
$\mathrm{var}(b_{1i})$	D11
μ	mu_1
$\xi^T = (\xi_1, \ldots, \xi_d)$	(xi_1,...,xi_d)
R	R
Z_i	z
σ^2	sigma_sq

```
/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=b1mean+z+b2mean*time;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
model response ~ general(loglik);
random z ~  normal(0,1) subject=subject;
predict z out=int_data;
ods output ParameterEstimates=parm_k0;
run;

proc transpose data=parm_k0 out=tparm_k0;
var Estimate;
id Parameter;
run;

data tparm_k0;
set tparm_k0;
test=1;
run;

data subjspec;
set int_data (keep=subject Pred StdErrPred last_obs);
where last_obs=1;
rename Pred=blup_1;
blup_var1=StdErrPred**2;
drop StdErrPred;
test = 1;
run;

data subjspec;
merge subjspec tparm_k0;
by test;
run;
```

```
data subjspec;
set subjspec;
G11 = sqrt(blup_var1);
beta_hat_1 = blup_1+b1mean;
run;

data subjspec2;
set subjspec (keep=subject G11 beta_hat_1);
run;

data data_set2;
merge data_set subjspec2;
by subject;
run;
```

2.2.2 Obtain Grid of Starting Values

Obtain starting values for mu_1 and R that give the same estimate of the mean and variance of the random effect as estimated with $K = 0$ over a grid of ξ values contained in the data set xi_matrix.

Starting values for $K = 1$.

```
data xi_matrix;
set xi_matrix;
test=1;
run;

data start_value2;
merge tparm_k0 xi_matrix;
by test;
run;

data start_value2;
set start_value2;
a0=sin(xi_1);
a1=cos(xi_1);
z1mean = 2*a0*a1;
z11var = a0**2+3*a1**2;
A=(z11var-z1mean**2);
R = sqrt(D11/A);
mu_1 = b1mean-R*z1mean;
run;

data start_value2;
/* Any other parameters for additional covariates should also be kept here*/
set start_value2 (keep=mu_1 b2mean R xi_1 sigma_sq);
run;
```

Starting values for $K = 2$.

```
data xi_matrix;
set xi_matrix;
test=1;
run;
```

```
data start_value2;
merge tparm_k0 xi_matrix;
by test;
run;

data start_value2;
set start_value2;
c1 = sin(xi_1);
c2 = cos(xi_1)*sin(xi_2);
c3 = cos(xi_1)*cos(xi_2);
a0=1.1944776*c1+-0.2705981*c3;
a1=c2;
a2=-0.2705981*c1+0.6532815*c3;
z1mean = 2*a0*a1+6*a1*a2;
z11var = a0**2+3*a1**2+15*a2**2+6*a0*a2;
A=(z11var-z1mean**2);
R = sqrt(D11/A);
mu_1 = b1mean-R*z1mean;
run;

data start_value2;
/* Any other parameters for additional covariates should also be kept here*/
set start_value2 (keep=mu_1 b2mean R xi_1 xi_2 sigma_sq);
run;
```

2.2.3 SNP Estimation

SNP Estimates with $K = 1$.

```
proc nlmixed data=data_set2 noad;
parms /data=start_value2;
pi = 2*arsin(1);
bounds -1.5707<= xi_1 <= 1.5707;
J = beta_hat_1-mu_1+G11*q;
z = J/R;
a0=sin(xi_1);
a1=cos(xi_1);
p_k=(a0+a1*z);
logden = log((p_k)**2);
re.dens = -0.5*(z**2);
q.norm = 0.5*(q**2);
log.det.iG = log(G11);
log.det.R.inv = log(1/R);

/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=mu_1+b2mean*time+R*z;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
if last_obs=1 then loglik=loglik+logden+re.dens+q.norm+log.det.iG+log.det.R.inv;
model cresponse ~ general(loglik);
random q ~ normal(0,1) subject=subject;
z1mean = 2*a0*a1;
z11var = a0**2+3*a1**2;
A=(z11var-z1mean**2);
estimate "b1mean" mu_1+R*z1mean;
estimate "D11" R**2*A;
run;
```

SNP Estimates with $K = 2$.

```
proc nlmixed data=data_set2 noad;
parms /data=start_value2;
pi = 2*arsin(1);
bounds -1.5707<= xi_1 <= 1.5707, -1.5707 <= xi_2 <= 1.5707;
J = beta_hat_1-mu_1+G11*q;
z = J/R;
c1 = sin(xi_1);
c2 = cos(xi_1)*sin(xi_2);
c3 = cos(xi_1)*cos(xi_2);
a0=1.1944776*c1+-0.2705981*c3;
a1=c2;
a2=-0.2705981*c1+0.6532815*c3;
p_k=(a0+a1*z+a2*z**2);
logden = log((p_k)**2);
re.dens = -0.5*(z**2);
q.norm = 0.5*(q**2);
log.det.iG = log(G11);
log.det.R.inv = log(1/R);

/* Any additional covariates used to describe the mean trajectory
 should be included here */

mu=mu_1+b2mean*time+R*z;

if censored=0 then loglik=log((1/(sqrt(2*pi*sigma_sq)))*exp(-(cresponse-mu)**2/(2*sigma_sq)));
if censored=1 then loglik=log(probnorm((cresponse-mu)/sqrt(sigma_sq)));
if last_obs=1 then loglik=loglik+logden+re.dens+q.norm+log.det.iG+log.det.R.inv;
model cresponse ~ general(loglik);
random q ~ normal(0,1) subject=subject;
z1mean = 2*a0*a1+6*a1*a2;
z11var = a0**2+3*a1**2+15*a2**2+6*a0*a2;
A=(z11var-z1mean**2);
estimate "b1mean" mu_1+R*z1mean;
estimate "D11" R**2*A;
run;
```

3 Additional Results from Simulations

Table 1: Proportion of subjects by number of non-censored observation for the four random effects densities consider in the simulation described in the text.

Distribution	0	1	2	3	4	5
Normal	0.001	0.086	0.239	0.146	0.094	0.434
t_5	0.002	0.077	0.225	0.162	0.111	0.423
Skewed	0.001	0.090	0.220	0.110	0.084	0.496
Bimodal	0.001	0.097	0.203	0.058	0.065	0.576

Table 2: Proportion of subjects by number of non-censored observation when the true random effects density was bimodal and the time points where data was collected was varied. $t_A = (0,1,2,3,4)^T$; $t_B = (0,1,2,3,3.5,4,4.5)^T$; $t_C = (-2,-1,0,1,2,3,4)^T$;

Time Points	0	1	2	3	4	5	6	7
t_A	0.001	0.097	0.203	0.058	0.065	0.576	0.000	0.000
t_B	0.001	0.096	0.203	0.049	0.029	0.033	0.052	0.538
t_C	0.000	0.000	0.001	0.096	0.205	0.066	0.091	0.541

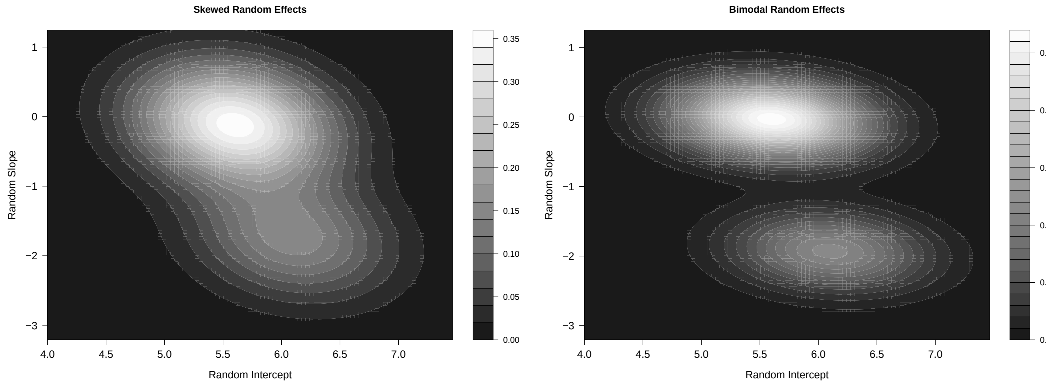

Figure 1: Contour plot of the random effects densities from the “skewed” and “bimodal” scenarios. In each case, the densities were a 70-30 mixture of two normals as described in the text.

Table 3: Proportion of time K was selected using AIC, HQIC, and BIC when SNP was used to estimate the random effects. The simulation included 500 data sets with 500 subjects each.

Distribution	Fit Criterion	K=0	K=1	K=2
Normal	AIC	0.562	0.134	0.304
Normal	HQIC	0.948	0.038	0.014
Normal	BIC	0.984	0.012	0.004
Skewed	AIC	0.000	0.020	0.980
Skewed	HQIC	0.002	0.046	0.952
Skewed	BIC	0.169	0.175	0.655
Bimodal	AIC	0.000	0.230	0.770
Bimodal	HQIC	0.000	0.636	0.364
Bimodal	BIC	0.000	0.900	0.100

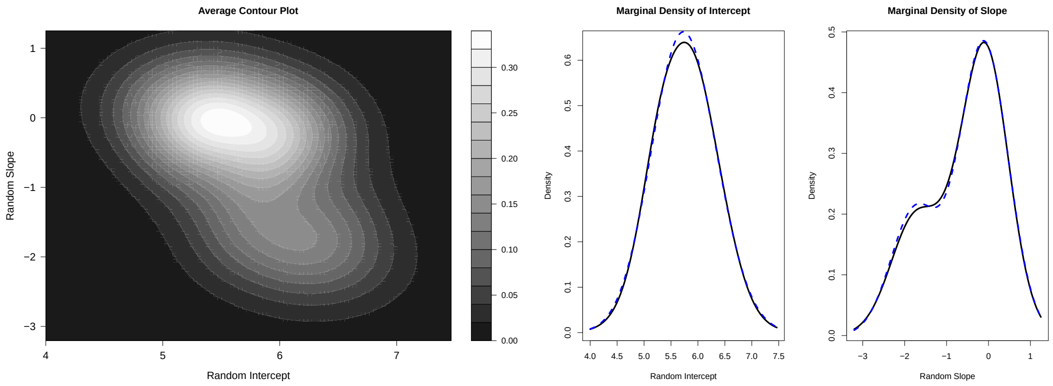

Figure 2: Average estimated contour and marginal density plots from fitting 500 data sets with "skewed" random effects where the density was assumed to follow the SNP density with K selected by HQIC. The blue dotted lines on the marginal density plots are the true marginal density.

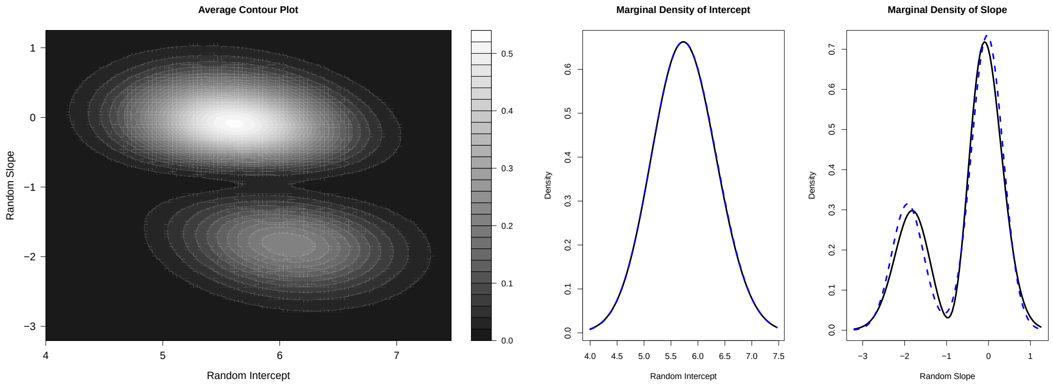

Figure 3: Average estimated contour and marginal density plots from fitting 500 data sets with "bimodal" random effects where the density was assumed to follow the SNP density with K selected by HQIC. The blue dotted lines on the marginal density plots are the true marginal density.

4 Additional Results from IDEAL Study

Table 4: Proportion of responses censored at each time point for the 811 patients with CT genotype in the IDEAL trial

Time (weeks)	0	2	4	12	Overall
Proportion	0.000	0.020	0.053	0.352	0.105

Table 5: Proportion of subjects by number of uncensored viral load measurements obtain for the 811 patients with the CT genotype in the IDEAL trial.

Uncensored Measurements	1	2	3	4
Proportion	0.045	0.083	0.312	0.551

Table 6: Parameter estimates, standard errors, and a 95 percent Wald-type confidence limits from fitting model (6.1) assuming the random effects followed the SNP density with $K = 2$.

Parameter	Estimate	Std Error	Lower Bound	Upper Bound
μ_1	5.940	0.035	5.872	6.008
μ_2	-0.758	0.019	-0.795	-0.722
ξ_1	-0.179	0.030	-0.237	-0.120
ξ_2	0.169	0.040	0.090	0.249
ξ_3	0.676	0.031	0.615	0.737
ξ_4	0.092	0.072	-0.050	0.234
ξ_5	0.594	0.069	0.459	0.729
R_{11}	0.422	0.019	0.385	0.459
R_{21}	0.096	0.011	0.074	0.117
R_{22}	0.229	0.009	0.212	0.246
σ^2	0.186	0.007	0.172	0.201
$\mathrm{E}(b_{0i})$	6.186	0.020	6.147	6.225
$\mathrm{E}(b_{1i})$	-0.393	0.014	-0.419	-0.366
$\mathrm{var}(b_{0i})$	0.200	0.017	0.167	0.233
$\mathrm{cov}(b_{0i}, b_{1i})$	0.080	0.009	0.061	0.098
$\mathrm{var}(b_{1i})$	0.135	0.011	0.114	0.155

5 Additional Simulations

For the $q = 1$ simulations we considered the random-intercept model

$$Y_{ij} = \beta_1 t_{ij} + \beta_0 g_i + b_{0i} + \epsilon_{ij}, \tag{14}$$

where $x_{ij} = (t_{ij}, g_i)^T$, $s_{ij} = 1$, g_i is equal to 0 or 1 with equal probability, $t_i = (t_{i1}, \ldots, t_{i5})^T = (0, 1, 2, 3, 4)^T$ is the same for all $i = 1, \ldots, m$, $\epsilon_{ij} \overset{iid}{\sim} N(0, \sigma^2)$, and b_{0i} are the subject-specific random intercepts, independent of ϵ_{ij}. For all simulations considered here, $\beta_0 = 0.5$, $\beta_1 = -0.6$, $\sigma^2 = 0.5625$, and b_{0i} were generated from distributions that were shifted and scaled so that $\mathrm{E}(b_{0i}) = 5.75$ and $\mathrm{var}(b_{0i}) = 3.0$.

The shifted and scaled random effects densities considered were: (1) normal; (2) t_5 distribution; (3) a 70-30 mixture of normal densities with mean components 6.61 (70 percent component) and 3.75, which gives a skewed density for b_{0i}; and (4) a 70-30 mixture of normal densities with mean components 4.75 (70 percent component) and 8.08, which produces a bimodal density for b_{0i}. Plots of these random effects densities are given in Figure (4). For the random effects densities that were a mixture of normal densities, the variance of each component were equal to each other. For each of the random effects distributions considered, 500 Monte Carlo data sets were generated with 500 subjects each and the lower limit of quantification was equal to 4 for all subjects and time points.

The proportion of subjects by number of non-censored observations for each of the random effects densities considered is given in Table (5). Across all scenarios, a majority of subjects had at least one censored observation. A fairly large percentage of subjects, approximately 10 percent in each scenario, had no observed responses.

For each data set, we obtained maximum likelihood parameter estimates using SAS Proc NLMIXED assuming that the random effects were normally distributed and assuming that they belonged to the class of SNP densities with $K = 0$, 1, or 2 and K selected using information criteria. When the random effects were assumed to be Gaussian (i.e. $K = 0$ for the SNP density), adaptive Gaussian quadrature was used to approximate the likelihood. When the random effects were assumed to follow the SNP density with $K > 0$, we used non-adaptive Gaussian quadrature to approximate the likelihood with the quadrature points centered at the empirical Bayes estimates of b_i derived from assuming the random effects were Gaussian. In all models fit, the number of quadrature points was selected adaptively to achieve a tolerance of 10^{-5} and dual quasi-Newton was used for optimization.

The results when Gaussian random effects were assumed are given in Table (8). When the random effects were normally distributed, parameter estimators are unbiased and, with the exception of the estimator for $\mathrm{E}(b_{0i})$, attain the stated level of confidence. However, when the random effects were non-normal, parameter estimators for $\mathrm{E}(b_{0i})$, $\mathrm{var}(b_{0i})$, and β_0 are all biased. The bias was largest when the random effects were bimodal, which, of the random effects densities considered, deviates most from normality. In spite of the the bias, coverage probabilities for $\mathrm{E}(b_{0i})$ and β_0 are adequate when the random effects were skewed and distributed as t_5.

The proportion of data sets that selected $K = 0$, 1, and 2 by information criteria is given in Table (9). When the random effects were Gaussian, $K = 0$ was selected for the vast majority of data sets for all three information criteria considered. When the random effects strongly departed from normality (skewed and bimodal random effects), $K = 0$ was rarely selected indicating the method's ability to detect departures from normality. Even when the random effects were distributed as t_5, which only slightly deviates from normality, HQIC selected $K = 0$ only 14.0 percent of the time. HQIC strikes a good balance between selecting $K = 0$ when the random effects are Gaussian and detecting subtle departures from normality.

Table (10) gives the parameter estimates and Figure (4) gives the estimated random effects density from fitting the SNP model with K selected by HQIC. For the skewed and bimodal random effects, parameter estimators are no longer biased. As a results of reducing the bias, coverage probabilities are closer to their stated level of confidence, and the SNP estimators are much more efficient. Because $K = 0$ is selected so frequently when the random effects are Gaussian, there is little loss in efficiency from considering a more flexible class of random effects under this scenario. However, because of the sizeable amount of subjects with no observed responses, SNP density estimation has a difficult time detecting the "fatter" tails in the t_5

density. For many data sets, the SNP density estimate of the random intercept places a second mode below the limit of quantification where there is little information to estimate that part of the density (Figure 5). As a result, the SNP estimator for var(b_{0i}) is much less efficient. To avoid the loss of efficiency, we recommend against relying solely on information criteria to select K. In addition to considering the information criteria, we suggest visually inspecting the density estimate for each K considered to avoid selecting values of K that have substantial mass where there is little information in the data set to estimate the density.

Table 7: Proportion of subjects by number of non-censored observation for $q = 1$ simulations.

Distribution	0	1	2	3	4	5
Normal	0.115	0.100	0.115	0.136	0.167	0.367
t_5	0.095	0.089	0.119	0.156	0.192	0.348
Skewed	0.134	0.090	0.089	0.111	0.167	0.408
Bimodal	0.085	0.147	0.176	0.156	0.111	0.325

Table 8: Simulation results when Gaussian random effects were assumed for all models regardless of the true distribution of the random effects. The simulation included 500 data sets with 500 subjects each and $q = 1$. MC Avg: Monte Carlo average of the parameter estimates; MC SD: Monte Carlo standard deviation of the parameter estimates; Avg SE: Average of the standard error estimates; CP: Monte Carlo coverage probability of the 95 percent Wald-type confidence intervals.

criteria	Distribution	E(b_{0i})	β_1	β_0	var(b_{0i})
Truth		5.750	-0.600	0.500	3.000
MC Avg	Normal	5.749	-0.600	0.498	2.991
MC Avg	t_5	5.763	-0.601	0.521	2.734
MC Avg	Skewed	5.778	-0.595	0.480	2.887
MC Avg	Bimodal	5.682	-0.609	0.529	3.387
MC SD	Normal	0.120	0.013	0.160	0.224
MC SD	t_5	0.110	0.013	0.155	0.318
MC SD	Skewed	0.117	0.012	0.157	0.218
MC SD	Bimodal	0.122	0.014	0.166	0.193
Avg SE	Normal	0.116	0.013	0.161	0.222
Avg SE	t_5	0.111	0.013	0.153	0.200
Avg SE	Skewed	0.114	0.013	0.158	0.219
Avg SE	Bimodal	0.123	0.014	0.170	0.242
CP	Normal	0.926	0.952	0.954	0.940
CP	t_5	0.954	0.950	0.950	0.618
CP	Skewed	0.930	0.932	0.960	0.896
CP	Bimodal	0.908	0.908	0.954	0.694

Table 9: Proportion of time K was selected using AIC, HQIC, and BIC for $q = 1$ simulations when SNP was used to estimate the random effects.

Distribution	Fit Criterion	K=0	K=1	K=2
Normal	AIC	0.786	0.121	0.093
Normal	HQIC	0.936	0.044	0.020
Normal	BIC	0.988	0.006	0.006
t_5	AIC	0.060	0.048	0.892
t_5	HQIC	0.140	0.096	0.764
t_5	BIC	0.292	0.131	0.577
Skewed	AIC	0.000	0.004	0.996
Skewed	HQIC	0.006	0.006	0.988
Skewed	BIC	0.030	0.010	0.960
Bimodal	AIC	0.000	0.056	0.944
Bimodal	HQIC	0.000	0.110	0.890
Bimodal	BIC	0.000	0.194	0.806

Table 10: Simulation results when SNP was used to estimate the random effects for $q = 1$ simulations. Models with $K = 0$, $K = 1$, and $K = 2$ were fit and K was selected using the HQIC. The simulation included 500 data sets with 500 subjects each. MC Avg: Monte Carlo average of the parameter estimates; MC SD: Monte Carlo standard deviation of the parameter estimates; Avg SE: Average of the standard error estimates; Ratio MSE: Ratio of the Monte Carlo mean square error between $K = 0$ and K selected by the HQIC.

criteria	Distribution	E(b_{0i})	β_1	β_0	var(b_{0i})
Truth		5.750	-0.600	0.500	3.000
MC Avg	Normal	5.749	-0.600	0.499	2.989
MC Avg	t_5	5.764	-0.599	0.513	2.771
MC Avg	Skewed	5.757	-0.600	0.493	3.012
MC Avg	Bimodal	5.745	-0.600	0.500	3.005
MC SD	Normal	0.120	0.013	0.161	0.230
MC SD	t_5	0.118	0.013	0.151	0.427
MC SD	Skewed	0.114	0.013	0.143	0.259
MC SD	Bimodal	0.098	0.014	0.097	0.163
Avg SE	Normal	0.116	0.013	0.159	0.222
Avg SE	other	0.108	0.013	0.143	0.266
Avg SE	other	0.113	0.013	0.145	0.242
Avg SE	other	0.096	0.014	0.097	0.169
CP	Normal	0.922	0.946	0.948	0.934
CP	t_5	0.920	0.940	0.932	0.668
CP	Skewed	0.942	0.946	0.960	0.914
CP	Bimodal	0.952	0.938	0.944	0.966
Ratio MSE	Normal	0.994	0.975	0.992	0.951
Ratio MSE	t_5	0.867	1.017	1.078	0.733
Ratio MSE	Skewed	1.112	1.140	1.225	0.899
Ratio MSE	Bimodal	2.015	1.401	3.005	7.023

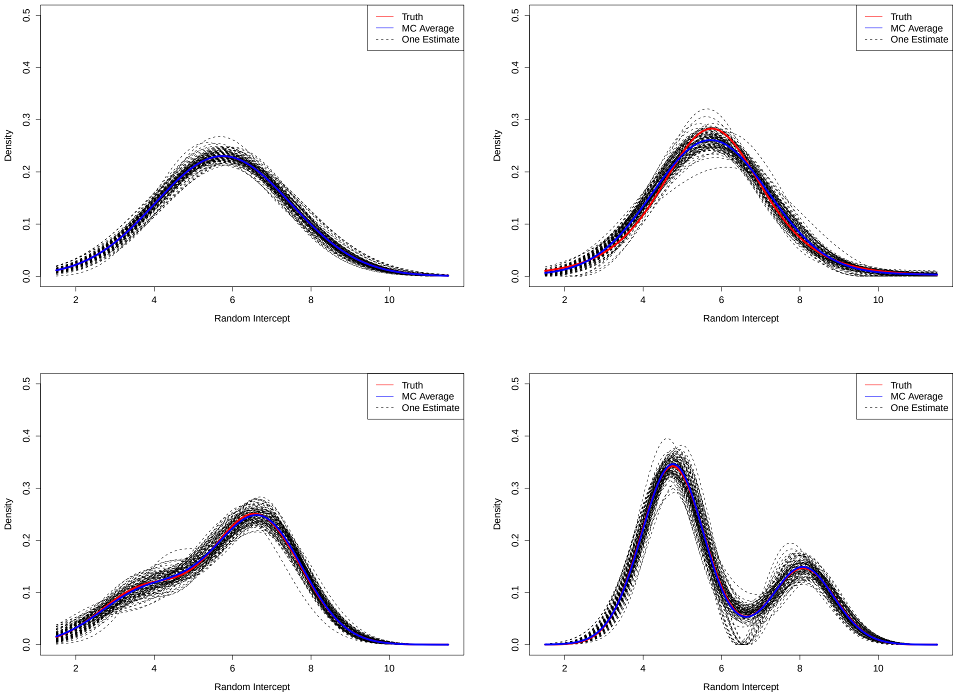

Figure 4: *Estimated random effects density for each of the four random effect densities considered (clockwise from upper left: normal, t_5, bimodal, and skewed) when SNP was used to estimate the random effects and K was selected using HQIC. The estimated density is plotted for 100 randomly selected Monte Carlo data sets. The truth is superimposed in red, and the average of all 500 Monte Carlo data sets is superimposed in blue.*

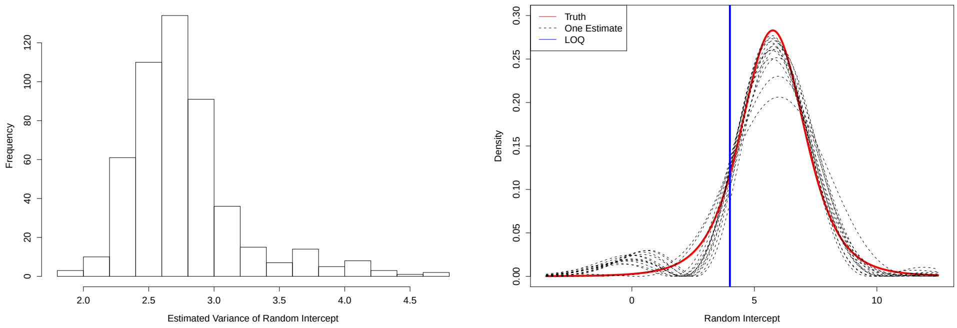

Figure 5: *Histogram of the estimated variance of the random effect when SNP was used to estimate the random effects density and the true random effect density was* t_5 *(left). The estimated random effects densities for those data sets where the estimate of* $var(b_{0i})$ *is greater than 4 are also plotted (right).*