Notation

In this chapter some basic definitions and notations of general linear mixed model used in this thesis are introduced.

Suppose we have $n$ observations, each observation is a binary process, which means $Y_i = 0$ or $1$, $i = 1, ..., n$ and we assume $Y_i$ are independent each other. The conditional expectation of response variable is:

$$P_i = E(Y_i|X_i, Z_i) = g^{-1}((\beta_1 X_{i1} + ... + \beta_m X_{im} + Z_i))$$ (1)

where $X_{ij}$ is $j^{th}$ fixed effect of $i^{th}$ observation, $j = 1, ..., m$. $\beta_j$ is $j^{th}$ fixed effect coefficient. $Z_i$ is $i^{th}$ random effect. In this paper, we assume $Z_i \sim N(0, \sigma^2)$, and there is only one random effect in the model. $g^{-1}$ is link function:

$$g^{-1}(x) = \frac{\exp(x)}{1 + \exp(x)}$$ (2)

In order to make the mathematics equations more clear, we expand these variables. Also, we only consider one random effect, and in order to use matrix form, we assume different client has same shape of data, same amount of observation in each random effect category, and it is easy to vary this condition to let different clients have different amount of observation. In the implementation, the observation number can be different for clients and random effect category. We use $I$ to denote number of clients, and $i \in [1, ..., I]$ is one instance of client. $T$ denote number of random effect categories, and $t \in [1, ..., T]$ is one instance of random effect categories. $M$ denote number of fixed effects, and $m \in [1, ..., M]$ is one instance of random effect categories. $P$ denote number of observation in one client one random effect category, and $p \in [1, ..., P]$ is one instance of observation.

In the next subsection, we are going to describe the likelihood problem in regular GLMM. After 2.2 we will describe how can we solve GLMM in both regular way and Fed GLMM.

Linear Mixed Model (LMM)

Linear Mixed Model (LMM) is an extension of the simple linear regression model for the data that are not completely independent, e.g., containing hierarchical structures [?]. LMM incorporates both fixed and random effects: a fixed effect does not vary within a subgroup of records (e.g., the patients sharing the same genotype in a case-control GWAS study) and can be modeled by a regression line across subgroups (e.g., three genotype classes: homozygous major, homozygous minor, and heterozygous, respectively), whereas a random effect is considered to be contributed by a random normal variate for records in each hierarchical category (e.g., smoking vs. non-smoking). Formally, an LMM is expressed as

$$y = \beta X + Zu$$ (3)
where $y$ is the outcome variable, $X$ represents the set of predictor variables for fixed effects, $\beta$ are the fixed effect regression coefficients, $Z$ represents the random effect variables complement to the fixed effects, and $u$ represents the random effect coefficients. Here, we consider the LMM with only one random effect, even though the privacy-preserving algorithms presented here can be extended to LMM with multiple random effects. Similar to the simple linear model, in an LMM, each fixed effect is measured by the coefficient (weight) of the variable. On the other hand, the random effect accounts for the randomness among the data in each of the $T$ categories ($T = 2$ for the random effect of smoking vs. non-smoking), which is modeled by the random variable $Z$ sampled from a normal distribution $N(0, \sigma)$.

**Likelihood**

Given parameters $\beta$ and $\sigma$, the likelihood is:

\[
L(\beta, \sigma|Y, Z) = \prod_{t=1}^{T} \prod_{p=1}^{P} p(Y_{tp}, Z_t|\beta, \sigma) \\
= \prod_{t=1}^{T} \prod_{p=1}^{P} f(Z_t) f(Y_{tp}|Z_t) \\
= \prod_{t=1}^{T} f(Z_t) \left[ \prod_{p=1}^{P} f(Y_{tp}|Z_t) \right] \\
= \prod_{t=1}^{T} f(Z_t) \left[ \prod_{p=1}^{P} f_{Y_{tp}|Z_t}(Y_{tp}|Z_t) \right] \\
= \prod_{t=1}^{T} f(Z_t) \left[ \prod_{p=1}^{P} f_{Y_{tp}|Z_t}(Y_{tp}|Z_t|\beta) \right] \\
= \left[ \prod_{t=1}^{n} f(Z_t) \right] \cdot \left[ \prod_{t=1}^{T} \prod_{p=1}^{P} f_{Y_{tp}|Z_t}(Y_{tp}|Z_t|\beta) \right]
\]

(4)
Then the log-likelihood is:

\[ l(\beta, \sigma | Y, Z) = \ln L(\beta, \sigma | Y, Z) = \sum_{t=1}^{T} \ln \left\{ f(Z_t) \left( \prod_{p=1}^{P} f(Y_{tp}|Z_t) \right) \right\} \]

\[ = \sum_{t=1}^{T} \ln f(Z_t) + \sum_{p=1}^{P} \ln f(Y_{tp}|Z_t) \]

\[ = \sum_{t=1}^{T} \left\{ -\frac{Z_t^2}{2\sigma^2} - \frac{1}{2} \ln (2\pi\sigma^2) + \sum_{p=1}^{P} [Y_{tp} \ln P_{tp} + (1 - Y_{tp}) \ln (1 - P_{tp})] \right\} \]

\[ = n \left[ -\frac{1}{2} \ln (2\pi\sigma^2) \right] - \sum_{t=1}^{T} \frac{Z_t^2}{2\sigma^2} + \sum_{t=1}^{T} \sum_{p=1}^{P} [Y_{tp} (X_{tp}\beta + Z_t) - Y_{tp} \ln (1 + e^{X_{tp}\beta + Z_t})] \]

\[ = n \left[ -\frac{1}{2} \ln (2\pi\sigma^2) \right] + \sum_{t=1}^{T} \left\{ -\frac{Z_t^2}{2\sigma^2} + \sum_{p=1}^{P} [Y_{tp} (X_{tp}\beta + Z_t) - \ln (1 + e^{X_{tp}\beta + Z_t})] \right\} \]  

(5)

The partial derivative of \( \beta_m \) is:

\[ \frac{\partial l}{\partial \beta_m} = \sum_{i=1}^{T} \sum_{j=1}^{P} Y_{ij} X_{m,ij} - \frac{X_{m,ij}e^{\beta_m X_{m,ij} + Z_i}}{1 + e^{\beta_m X_{m,ij} + Z_i}} \]

(6)

where \( m \in [0, 1, ..., M] \), so the \( \beta_m \) is \( m^{th} \) fixed effect coefficient. \( X_m \) is the \( m^{th} \) fixed effect.

In order to optimize fixed effects(8), we need a EM algorithm, which means we need to use Metropolis Hasting sampling algorithm to find the best random effect \( Z \) under the current fixed effect \( \beta \), and use Newton-Raphson algorithm to find the best fixed effect under the current random effect \( Z \). So the whole procedure is a EM algorithm. In next subsection we will talk about the detail of Metropolis Hasting Algorithm in Fed GLMM, and section we will talk about Newton-Raphson algorithm in Fed GLMM in later section.

**Collaborative Metropolist-Hasting**

In this stage, current \( \beta \) has already known. We then use current \( \beta \) to estimate random effect. In order to sample \( Z \), we use Metropolis Hasting Algorithm(MH) to sample it. The regular MH states that, we choose \( h_Z(z) \) as the candidate distribution, mostly we use normal distribution as candidate distribution. We draw \( z^* \) and accept \( z^* \) as \( Z_i \) with probability:

\[ A_{ui,Y_i} (z, z^*) = \min \left\{ 1, \frac{f(z^*) h(z|z^*)}{f(z) h(z^*|z)} \right\} = \min \left\{ 1, e^{\sum_{j=1}^{P} Y_{ij}(z^*-z)} \prod_{j=1}^{P} \frac{1 + e^{X_{ij}\beta + z}}{1 + e^{X_{ij}\beta + z^*}} \right\} \]  

(7)

Where \( A \) In another word, \( A \) is like a filter to determine whether update to \( z_i^* \) or not. If we take log of the RHS equation in (10):

\[ \text{RHS} = \sum_{j=1}^{P} Y_{ij} (z^* - z) + \sum_{j=1}^{P} \log(1 + e^{X_{ij}\beta + z}) - \sum_{j=1}^{P} \log(1 + e^{X_{ij}\beta + z^*}) \]

(8)

Which we can separate (11) as sum of the intermediary results from different clients:

\[ \text{RHS} = \sum_{k=1}^{m} \left\{ \sum_{j=1}^{P} Y_{k,ij} (z^* - z) + \sum_{j=1}^{P} \log(1 + e^{X_{k,ij}\beta + z}) - \sum_{j=1}^{P} \log(1 + e^{X_{k,ij}\beta + z^*}) \right\} \]

(9)
So for Metropolis Hasting sampling in Fed GLMM, there are many data communication between clients and center. In the beginning, center set a initial random effect variance \( \sigma_0 \), and generate \( Z = Z_1, ..., Z_T \) from \( N(0, \sigma_0) \), send \( Z_1, ..., Z_T \) to each client. Each client use this Z compute intermediary results, send back to center. Center sample next group of random effect, denote as \( Z^* \), and use those intermediary results from clients to compute A in order to decide keep or replace which element in Z. Formally, in \( k^{th} \) client, it need to compute intermediary results of A, where A is a \( P \) dimensional array. Then \( i^{th} \) element of A for \( k^{th} \) client is:

\[
A_{k,j} = \sum_{j=1}^{P} Y_{k,ij} (z^* - z) + \sum_{j=1}^{P} \log(1 + e^{X_{k,ij}\beta + z}) - \sum_{j=1}^{P} \log(1 + e^{X_{k,ij}\beta^* + z^*})
\]  

(10)

center sum up all clients to get the A:

\[
A = \sum_{k=1}^{m} A_k
\]

(11)

Center then use this A to chose whether we need to update Z or not. Do these step in n iterations to receive samples of Z. The pseudo code show below:

\[
\sigma \text{ is the variance of previous MH samples}
\]

**Initialization:**
Z = sample T numbers from \( N(0, \sigma) \), 
i = 1 , k = 1, 
\( Z_{MH} = [] \)

while \( i < n \) do
    newZ = sample T numbers from \( N(0, \sigma) \)
    while \( k < m \) do
        compute \( A_k \); send to center; 
k++
    end
    center sums up all \( A_k \) : \( A = \sum_{k=1}^{m} A_k \)
    \( Z = \text{ifelse(log(runif(T)) > A)} \), \( Z, \text{newZ} \)
    \( Z_{MH}.\text{add}(Z) \)
i++
end
return \( Z_{MH} \)

**Algorithm 1:** MH sampling in GLMM

In next subsection we will describe how to do Optimization with Monte Carlo Newton-Raphson in Fed GLMM.

**collaborative Newton-Raphson**

Monte Carlo Newton Raphson in Fed GLMM can compute the fixed effect when we fixed random effect. Basicly, we need to compute the fist order derivative \( f' \) and second order derivative \( f'' \) of fixed effects. For each step \( \beta \) update \( f'/f'' \) until converge.

Use from section 2.2 we know the partial derivative of \( \beta \) is :

\[
\frac{\partial l}{\partial \beta_c} = \sum_{i=1}^{T} \sum_{j=1}^{P} \left[ Y_{ij} X_{c,ij} - X_{c,ij} e^{\beta_c X_{c,ij} + Z_i} \right]
\]

(12)

where \( c \in [0, 1, ..., m] \), so the \( \beta_c \) is \( c^{th} \) fixed effect coefficient. \( X_c \) is the \( c^{th} \) fixed effect.

and for second order derivative, element of \( s^{th} \) column \( q^{th} \) row of Hessian Matrix \( H_{s,q} \) is:

\[
H_{s,q} = -\sum_{i=1}^{T} \sum_{j=1}^{P} \frac{X_{s,ij} X_{q,ij} e^{\beta_s X_{s,ij} + \beta_q X_{q,ij} + Z_i}}{(1 + e^{\beta_s X_{s,ij} + \beta_q X_{q,ij} + Z_i})^2}
\]

(13)
For the Fed GLMM, first let’s see the first order derivative.

\[
\frac{\partial l}{\partial \beta_c} = \sum_{i=1}^{T} \sum_{j=1}^{P} \left[ Y_{ij} X_{c,ij} - \frac{X_{c,ij} e^{\beta_c X_{c,ij} + Z_i}}{1 + e^{\beta_c X_{c,ij} + Z_i}} \right]
\]

(14)

So for each client, compute a intermediary result, and then send to center, center sum them up, then we can receive first order derivative. For example, \( k^{th} \) client compute following:

\[
\sum_{i=1}^{T} \sum_{j=1}^{P} \left[ Y_{k,ij} X_{k,c,ij} - \frac{X_{k,c,ij} e^{\beta_c X_{k,c,ij} + Z_i}}{1 + e^{\beta_c X_{k,c,ij} + Z_i}} \right]
\]

(15)

Where \( X_{k,c,ij} \) is the fixed effect variable in \( k^{th} \) client \( c^{th} \) fixed effect, \( i^{th} \) category of random effect \( j^{th} \) observation. and \( Y_{k,c,ij} \) defined in the same fashion.

Then center sum them up, to get the first order derivative

For the \((s,q)^{th}\) element of Hessian matrix,

\[
H_{s,q} = -\sum_{i=1}^{T} \sum_{j=1}^{P} X_{s,ij} X_{q,ij} \frac{e^{\beta_s X_{s,ij} + \beta_q X_{q,ij} + Z_i}}{(1 + e^{\beta_s X_{s,ij} + \beta_q X_{q,ij} + Z_i})^2}
\]

(16)

So similar with first order derivative, for each client, compute a intermediary result, and then send to center, center sum them up, then we can receive first order derivative. For example, \( k^{th} \) client compute following:

\[
\sum_{i=1}^{T} \sum_{j=1}^{P} X_{k,s,ij} X_{k,q,ij} e^{\beta_s X_{k,s,ij} + \beta_q X_{k,q,ij} + Z_i}
\]

(17)

Then center sum them up, to get the \((s,q)^{th}\) element of Hessian Matrix.

**Result Appendix**
Figure 1: 50 SNPs of simulated data

Figure 2: 100 SNPs of simulated data
Figure 3: 50 SNPs of GWAS

Figure 4: 100 SNPs of GWAS