Bayesian Linear Mixed Models for biomedical data analyses.

Shamsi Abdurakhmanova

February 2, 2022

1 Introduction

[Code and other related materials].

Linear models are the most commonly used statistical methods for data prediction and inference in science, business and medicine. Such popularity is due to simplicity and interpretability of a linear models.
Multiple (meaning multiple predictors) linear model is defined as a map

$$\begin{aligned} Y &= \beta_0 + \mathbf{X}\beta + \epsilon \\ \epsilon &\sim \mathcal{N}(0, \mathbf{I_m}\sigma^2) \end{aligned} \tag{1}$$

or equivalently

$$\begin{aligned} \mu &= \beta_0 + \mathbf{X}\beta \\ Y &\sim \mathcal{N}(\mu, \mathbf{I_m}\sigma^2) \end{aligned} \tag{2}$$

where $\mathbf{X}$ is $m \times n$ feature matrix, Y is a $m \times 1$ vector of observations, β_0 is an intercept and β is a $n \times 1$ vector of coefficients. Error $m \times 1$ vector is denoted as ϵ.
m is a number of observations and n is a number of predictors.

Ordinary least squares method can be used to estimate model parameters

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T Y \tag{3}$$

This type of regression is also called General Linear Model (GLM). The are several assumptions for a linear regression[1]:

- Linearity of relationship between predictors and response variable
- Independence of outcome/ response variables
- Data (observations) is approximately normally distributed
- The prediction errors are constant over the entire data (homoscedasticity)
- Independence and normality of the prediction errors (residuals)
- Predictors are not correlated (avoid multicollinearity)

Experimental design in biology and biomedicine usually complex and number of observed samples is low, which leads to various problems in an analyses of collected data. Experiments often contain multiple treatments, repeated measures, various grouping factors (blocks) and worst of all, unknown confounding variables.

Consider a typical experiment in the field of drug development (Figure 1, blue box): mice are treated with either saline (control group) or drug (treatment group). Groups are formed from female and male mice. Testing done on 3 batches of animals. Several samples (measurements) are collected from each mice.
We can do the same experiment as described above, but on a different line of mice - wild type and

gene modified (Figure 1). Thus, by adding more predictors and/or groups, we can design increasingly more complex experiments.

We can identify following variables in the described experiment:

- **response (dependent) variable** - collected samples/measurements
- **predictors (independent) variables** - genotype, treatment, sex

In addition, we can identify factors, which divide the data into groups or clusters or blocks: batches (batch 1-3) and repeated measurements (sample 1-3). We could ignore grouping factors and apply multiple linear regression to model the response variable, but there is a chance that we will miss important information or fail to detect spurious relationships.

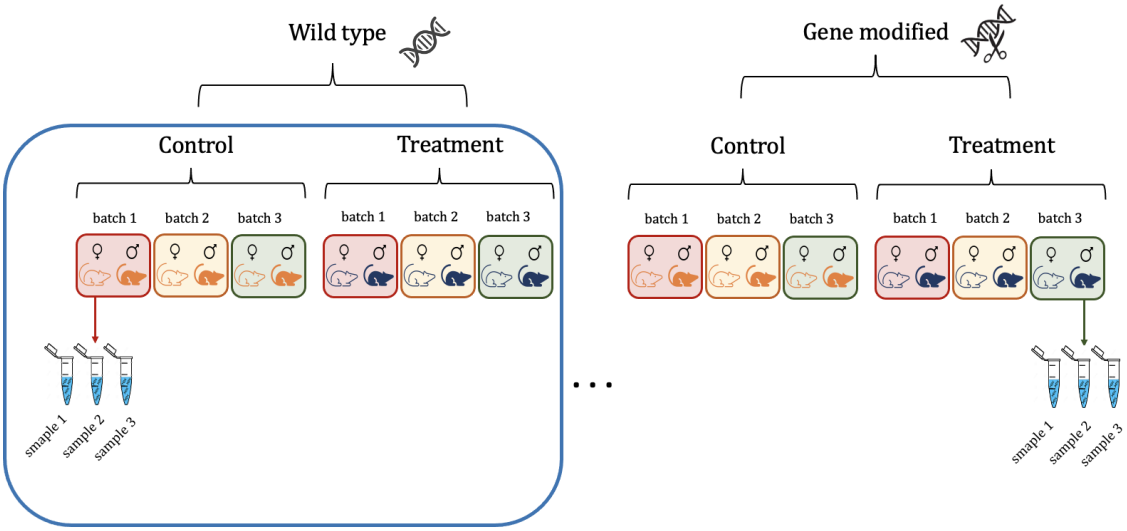


Figure 1: Example of experimental design in the field of biomedicine.

For example, consider the result of a hypothetical experiment[2] depicted on Figure 2: each group consist of samples from two different batches and we can clearly see the effect of a batching on response variable. Ignoring the batching effect would lead to wrong conclusions.

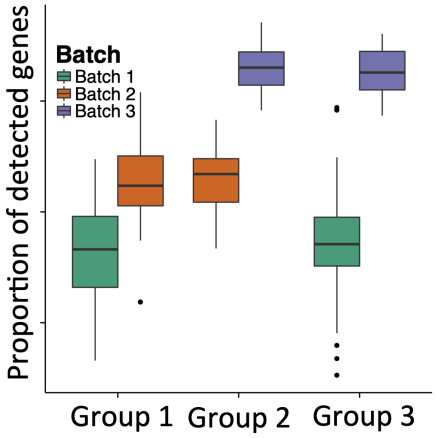


Figure 2: Batch effect.

It is possible to include batching and repeated measures as predictors into multiple linear model or use analyses of variance methods (e.g. repeated measures ANOVA), but there are several downsides to this approach.

Firstly, the number of groups/block can be very large and addition of a new group requires re-fitting the model from scratch. Imagine that you study light-dependent circadian rhythms and you suspect that lighting condition in animal facility is different in different rooms. You want include information about room (e.g. room number) in your data analyses. Let's say you had your animals in three different rooms and you include independent factor "room id" as a categorical variable with three levels. Each time new animals arrive, they are put in a new room and you need to re-define variable "room id" and add more categories/levels. In addition, if there are few animals per each room and there are many rooms (i.e. the dataset is small compare to number of predictor levels), it will be difficult to reliably estimate the "room effect".

Secondly, these methods require that data is balanced and follows multiple assumptions about data and prediction error distributions. Thus, they are rather restrictive for biological data, which normally is unbalanced, has few collected samples, contains missing values and often very noisy (high variation or low signal to noise ratio).

One popular method to approach these problems is use of Linear Mixed Models (LMM) which differentiate between fixed and random effects. **Fixed** effects are predictors which are constant, "fixed" on a population level, while **random** effects are "fixed" on a group/ block level, but not on the population level. For example, imagine you want to predict final grade of students, based on some collected information about students' performance during a year. You suspect that different schools might have different "baseline" performance and by pooling (averaging) data from different schools you may get model with poor prediction ability:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_i + \epsilon_i \\ \epsilon_i &\sim \mathcal{N}(0, \sigma^2) \end{aligned} \tag{4}$$

To account for this "school effect" you could allow intercept of a linear model vary between different schools. The problem then is formulated as follows:
Varying intercept model

$$\begin{aligned} y_{ij} &= \beta_0 + u_j + \beta_1 x_{ij} + \epsilon_{ij} \\ u_j &\sim \mathcal{N}(0, {\sigma_u}^2) \\ \epsilon_{ij} &\sim \mathcal{N}(0, {\sigma_\epsilon}^2) \end{aligned} \tag{5}$$

where β_0 is a population level intercept, u_j is a random intercept for each jth school, β_1 is a population level slope, y_{ij} is a final grade of an ith student of jth school after introduction of a new teaching method and ϵ_{ij} is an error term.

In general, a linear mixed effects model extends linear regression as[3]:

$$\begin{aligned} Y &= \beta_0 + \mathbf{X}\beta + \mathbf{Z}\eta + \epsilon \\ \eta &\sim \mathcal{N}(0, \mathbf{I_q}{\sigma_\eta}^2) \\ \epsilon &\sim \mathcal{N}(0, \mathbf{I_m}{\sigma_\epsilon}^2) \end{aligned} \tag{6}$$

where there is still a slope vector β, intercept β_0, and random noise ϵ. In addition, there is a term $\mathbf{Z}\eta$, where $\mathbf{Z}$ is a features matrix and η is a vector of random slopes; η is normally distributed with variance ${\sigma_\eta}^2$. $\mathbf{Z}$ is formed by partitioning the original $m \times n$ features matrix in terms of a new $m \times p$ matrix $\mathbf{X}$ (design matrix) and $m \times q$ matrix $\mathbf{Z}$ (block matrix), where $p + q = n$: this partition allows to model the features separately using the fixed effects β and the random effects η respectively.

We say the random effects are effects that vary across the population (although they may be constant across subpopulations). In particular, because the random effects have mean 0, the data label's mean is captured by $\beta_0 + \mathbf{X}\beta$. The random effects component $\mathbf{Z}\eta$ captures variations in the data due to presence of groups/ blocks/ clusters.

LMM parameters are estimated with maximum likelihood (ML) or Restricted Maximum Likelihood (REML) approaches[4] and can be computed, e.g. with lme4 R package [5].

LMMs can be implemented as Bayesian hierarchical models, where instead of ML/REML, posterior distribution is estimated with use of sampling techniques (e.g. Markov chain Monte Carlo). In this report we implement and compare LMM and Baysian LMM with use of lme4 and Rstan R packages on a dataset typical for experimental biomedicine.

2 Data

Data was collected during my doctoral study at University of Helsinki. The aim of the experiment was to estimate the effect of drug ciproxifan on electrocorticogram of laboratory mice. Ciproxifan is known to affect specific brain receptors and induce wakefulness in laboratory animals and humans, as measured by electrocorticogram (ECoG) or electroencephalogram.
Experiment has been done in two parts. First, mice (n=18) were treated either with saline or cipoxifan 10mg/kg and ECoG recorded for few hours. After 2 week wash-out period, same mice were treated either with saline or cipoxifan 3mg/kg and ECoG recorded. Mice were either wild-type (wt) or Gabrd knockout (ko). Gabrd knockout mice are gene modified (transgenic) mice with deletion of δ-subunit of GABAA receptor. Wild-type mice are the control group for ko mice, with intact δ-GABAA receptor. Three mice have to be excluded from 2nd part due to technical or health reasons. Schematic illustration of an experiment is found in Figure 3.

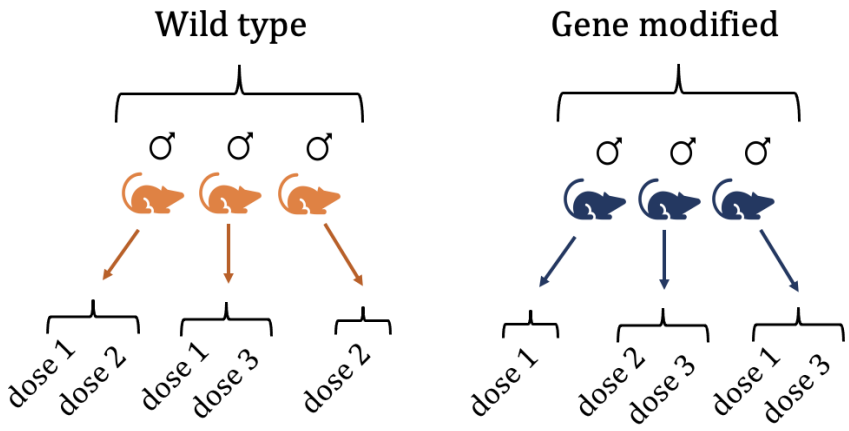


Figure 3: Experimental design.

Each mouse received one of these dose combinations:

- 0+3 (wt n=4, ko n=2)
- 10+3 (wt n=1, ko n=2)
- 10+0 (wt n=3, ko n=3)
- 0+NA (ko n=2)
- 10+NA (wt n=1)

Collected ECoG recordings were pre-processed and manually labeled for the following stages: wake state (WAKE), non rapid eye movement sleep (NREM), rapid eye movement sleep (REM).

	id	NREM	REM	WAKE	Genotype	Treatment
0	16	53.423744	0.266785	46.309471	1	0.0
1	15	60.544677	9.025433	30.429890	1	0.0
2	17	20.456600	0.452080	79.091320	1	10.0
3	9	26.550079	0.794913	72.655008	0	10.0
4	12	42.585377	0.664680	56.749943	0	0.0

Figure 4: Dataset.

The dataset consist of 33 measurements from 18 mice. Each measurement is consist of relative amount of WAKE, NREM, REM stages (Figure 4). Each measurement has corresponding value of factors "Genotype" and "Treatment". Genotype factor is a dummy variable coded as follows: 0 for wt and 1 for ko. Treatment is continuous variable (dose in mg/kg) with values 0,3,10.

The goal of the data analyses is formulated as follows: predict relative amount of WAKE given information about Treatment (dose) and Genotype (wt or ko). Visual inspection of data indicates that higher dose of drug result in increased WAKE amount (Figure 5). Find full (exploratory data analyses) EDA in jupyter notebook "EDA.ipynb".

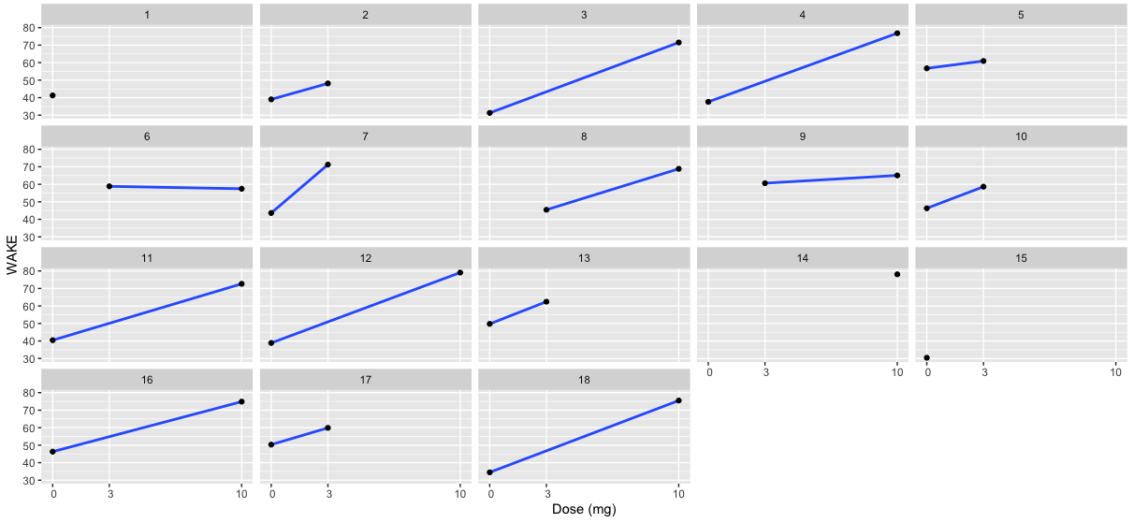


Figure 5: Relative wake amount for each subject.

By combining the data for wt and ko mice, it seems that there might be interaction of factors Treatment:Genotype (see Figure 6).

First, we implement this problem as Fixed effects regression with two predictors: Treatment (dose) and Genotype (wt or ko). As we also have unbalanced repeated measurements, we will add this information by using Mixed effects model and compare results.

We will use Maximum likelihood and Bayesian approach to fit Linear Fixed and Mixed models.

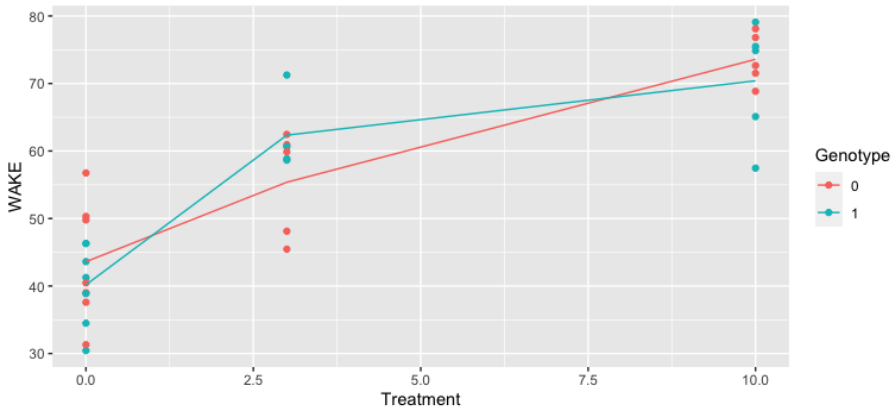


Figure 6: Relative wake amount for wt and ko mice.

3 Fixed effects regression

3.1 Estimating regression parameters with MLE approach.

We fit fixed effects linear model as was discussed above, but with addition of interaction term TreatmentxGenotype:

$$Wake_i = \beta_0 + Treatment_i\beta_1 + Genotype_i\beta_2 + Treatment_i * Genotype_i\beta_3 + \epsilon_i$$
$$\epsilon_i \sim \mathcal{N}(0, \sigma)$$

First we will use lm() function which estimates model parameters $\beta_0, \beta_1, \beta_2, \beta 3$ with maximum likelihood estimation (MLE). The estimated values ($\pm st.error$) are in Table 1 **??**. The fitted model plotted in Figure 7 with shaded area being 95% confidence level (CI) interval (95% CI implies that if the entire study were repeated identically ad infinitum, 95% of such confidence intervals formed in this manner will include the true value).

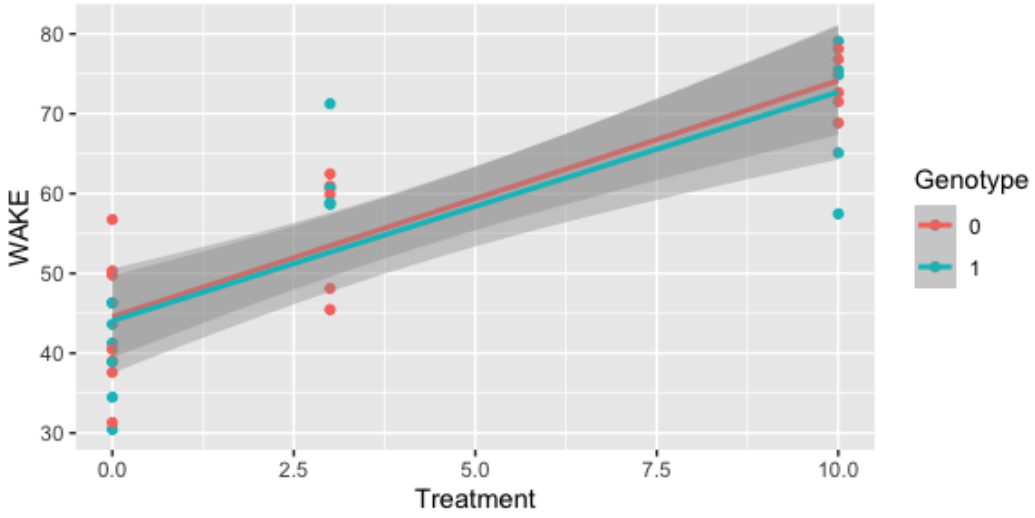


Figure 7: Fixed effect regression with MLE.

The lm() output (see Listing 1) includes parameters' estimates, standard error (square root of a variance), t value and value $Pr(>|t|)$ [6].
t value is the value of the t-statistic for testing whether the corresponding regression coefficient is different from 0 (t=estimate/st.error).

$Pr(>|t|)$ is the p-value for the hypothesis test for which the t value is the test statistic. It tells you the probability of a test statistic at least as unusual as the one you obtained, if the null hypothesis were true. In this case, the null hypothesis is that the true coefficient is zero; if that probability is low, it's suggesting that it would be rare to get a result as unusual as this if the coefficient were really zero.

The lm() output shows significant P-value for intercept and factor Treatment, but not for factor Genotype or Treatment:Genotype interaction.

Given fitted model we can check some assumptions about the model. We can use R function plot(fit) to automatically plot subplots (Figure 8):

- residuals vs fitted values

This plot shows if residuals have non-linear patterns. If residuals are equally spread around a horizontal line without distinct patterns, it is a good indication that there are no non-linear relationships. Based on the plot, it seems that there are some non-linearity in dose-response curve.

- normal Q-Q (quantile-quantile) plot

This plot helps to assess if a set of data plausibly came from some theoretical distribution. As we assume that response variable is normally distributed, we use Normal Q-Q plot. There are some outliers visible in the plot, but generally, normality assumption seems to hold.

- scale-location

This plot provides information about homoscedasticity. Homoscedasticity means that the variance of the residuals remains constant and does not correlate with any independent variable. In unproblematic cases, the graphic shows a flat line, which is approximately the case for our data.

- residuals vs leverage

This plot helps us to find influential cases (i.e., observations). The influential cases are the points outside of a dashed line, Cook's distance. When cases are outside of the Cook's distance (meaning they have high Cook's distance scores), the cases are influential to the regression results. The regression results can change greatly if we exclude those cases. In our case, there are no influential observations.

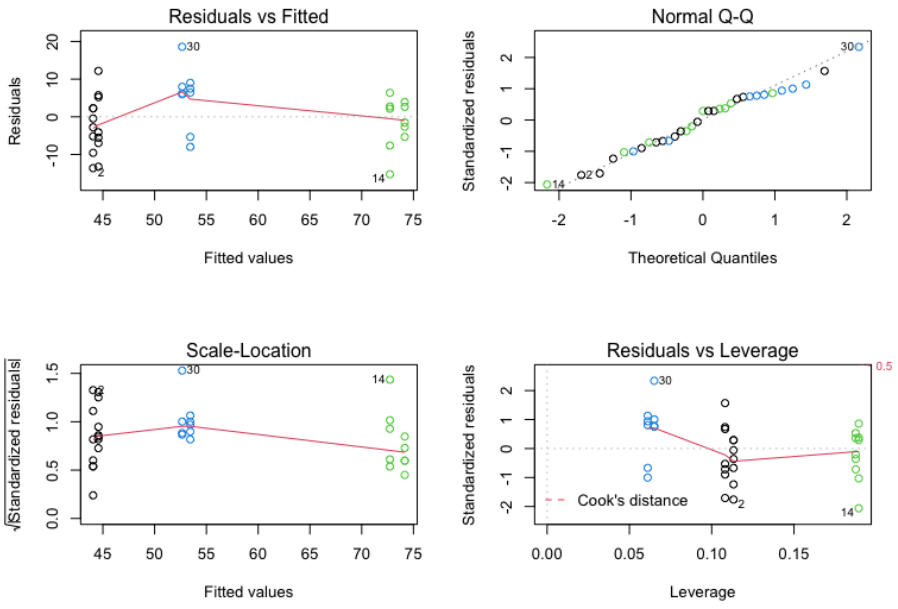


Figure 8: Diagnostic Plots for Fixed Effects Regression

3.2 Estimating regression parameters with Bayesian approach

In this section we employ Bayesian approach for fitting fixed effect regression. First, we can verify that lm() results are similar to maximum a posteriori estimation (MAP) in Bayesian hierarchical model with flat (uninformative) priors (see Listing 2). The model is therefore:

$$
\begin{gathered}
Wake_i = \beta_0 + Treatment_i\beta_1 + Genotype_i\beta_2 + Treatment_i * Genotype_i\beta_3 + \epsilon_i \\
\beta \sim U(-\inf, +\inf) \\
\epsilon_i \sim \mathcal{N}(0, \sigma) \\
\sigma \sim U(0, +\inf)
\end{gathered}
$$

Next, we change flat priors to weakly informative priors:

$$
\begin{gathered}
Wake_i = \beta_0 + Treatment_i\beta_1 + Genotype_i\beta_2 + Treatment_i * Genotype_i\beta_3 + \epsilon_i \\
\beta \sim \mathcal{N}(0, 100) \\
\epsilon_i \sim \mathcal{N}(0, \sigma) \\
\sigma \sim \mathcal{N}(0, 10)
\end{gathered}
$$

Location parameter for all β's is 0 and scale is 100 (see Listing 3).
We choose normal prior with zero mean for coefficients β's, starting from the initial guess that fixed factors have small effect, thus coefficients magnitudes will be around zero. We do not have any specific knowledge about coefficients, thus normal distribution is weak enough prior. Although normal priors are not advised for scale parameters for hierarchical models [7], we will use half-normal distribution, as the results with different priors (half-cauchy and t-distribution, data not shown) were showed to be similar in this case.

We can visually check that Markov chains (4 chains, each with iter=2000; warmup=1000) converged (Figure 9) and compute $\hat{R}$ for quantitative evaluation of convergence [9]. In general, $\hat{R}$ estimates the between- and within-chain estimates for model parameters to perform convergence diagnostic. If chains have not mixed well (ie, the between- and within-chain estimates don't agree), $\hat{R}$ is larger than 1. $\hat{R}$ values for all parameters are 1 (see Listing 3).
Bulk ESS and Tail ESS are crude measures of effective sample size for bulk and tail quantities respectively. ESS values larger than 100 per chain is considered good, thus, for 4 chains ESS values should be larger than 400. We can print out summary of stan fit with monitor() function and see that ESS values are larger enough. There were no divergent transitions or any kind of other warning messages.

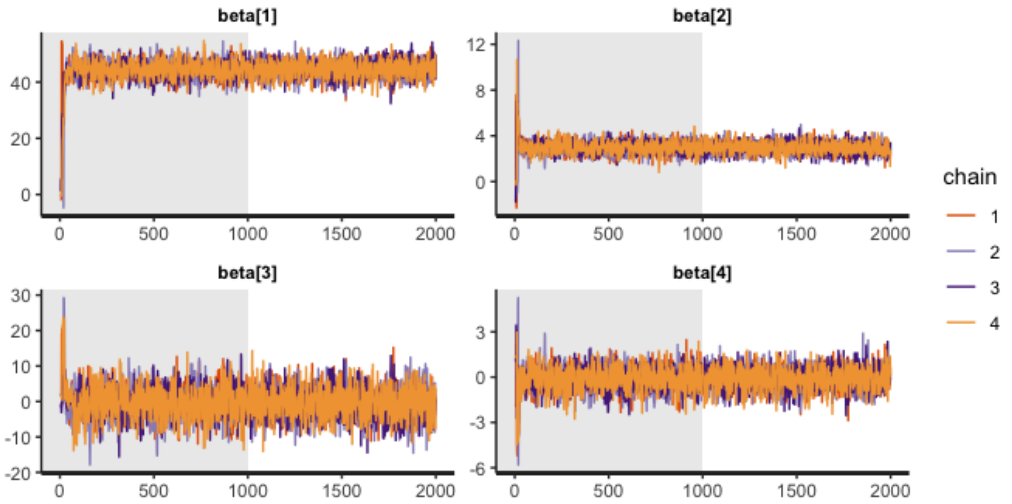


Figure 9: Markov chains, Fixed effects regression.

4 Mixed effects Regression

In the first section we performed analyses while violating assumption about conditional independence of residuals, as dataset contains repeated measurements. In order to account for repeated measurements we add random factor 'id' where animals' ids are stored. We introduce varying intercept u which value will depend on animal id.

4.1 Estimating regression parameters with MLE approach.

We formulate the problem as follows (see Listing 4):

$$
\begin{gathered}
Wake_{ij} = \beta_0 + u_{0j} + Treatment_{ij}\beta_1 + Genotype_{ij}\beta_2 + Treatment_{ij} * Genotype_{ij}\beta_3 + \epsilon_{ij} \\
u_{0j} \sim \mathcal{N}(0, \sigma_u) \\
\epsilon_{ij} \sim \mathcal{N}(0, \sigma_\epsilon)
\end{gathered}
$$

The estimated parameters values are still very similar to previous two models (see Table 1).

4.2 Estimating regression parameters with Bayesian approach

We formulate the problem as follows (see Listing 5):

$$
\begin{gathered}
Wake_{ij} = \beta_0 + u_{0j} + Treatment_{ij}\beta_1 + Genotype_{ij}\beta_2 + Treatment_{ij} * Genotype_{ij}\beta_3 + \epsilon_{ij} \\
\beta \sim \mathcal{N}(0, 100) \\
u_{0j} \sim \mathcal{N}(0, 10) \\
\epsilon_{ij} \sim \mathcal{N}(0, \sigma) \\
\sigma \sim \mathcal{N}(0, 10)
\end{gathered}
$$

Based on visual inspection and $\hat{R}$ values (see Listing 5) we can conclude that Markov chains are converged. There were no warning messages about ESS values or divergent transitions.

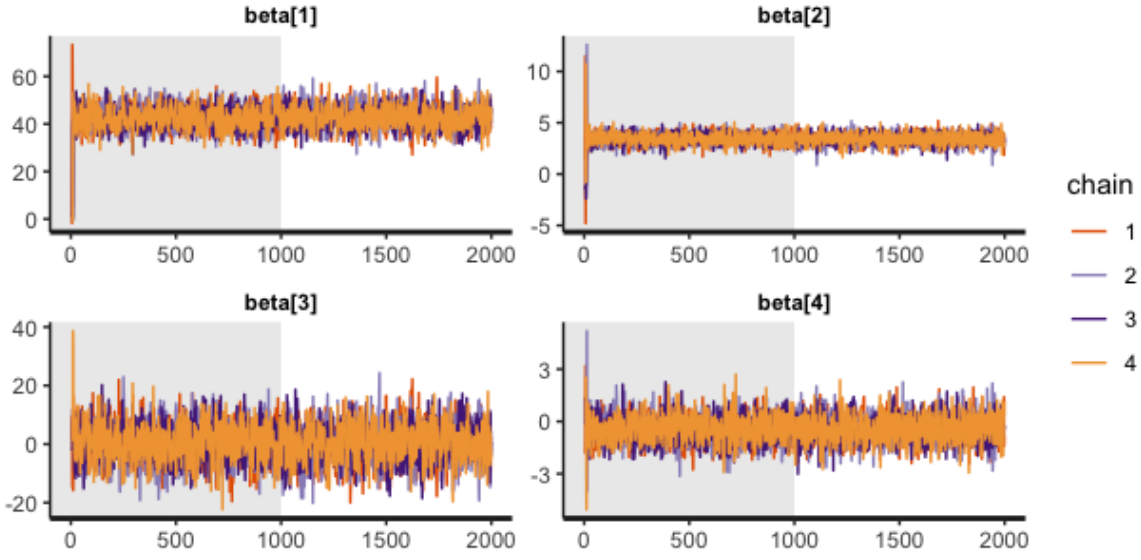
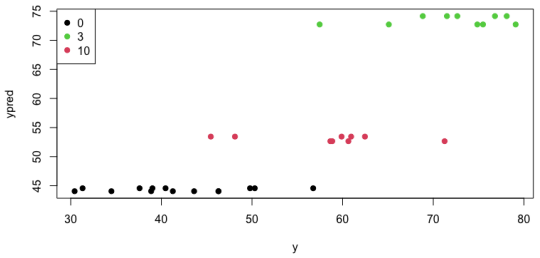


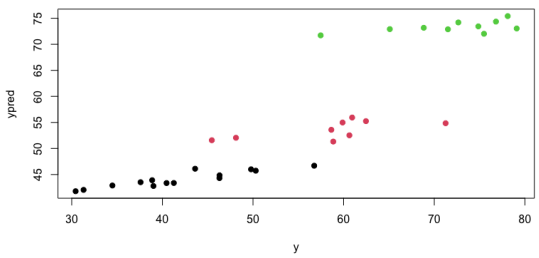
Figure 10: Markov chains, Mixed effects regression.

5 Model predictions

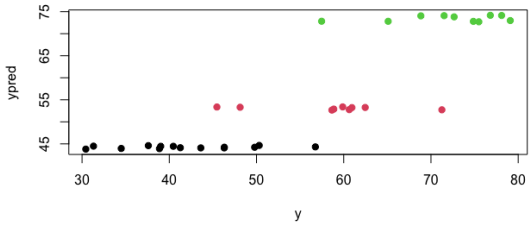
We can estimate model fit by plotting response variable against predicted values (for Bayesian LMM mean estimate is used). The color of the dots is related to treatment dose.
Clearly, both models with a random intercept fit data better, than fixed effects models. Bayesian LMM with random intercept fits data better than LMM with random intercept fitted with MLE approach. Mean squared errors values confirmed this conclusion.



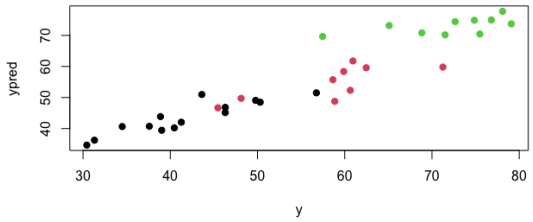
(a) Fixed effects, lm()



(b) Mixed effects, lmer()



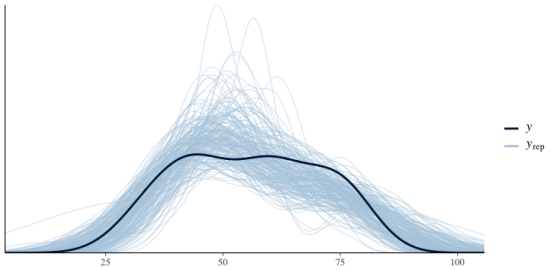
(c) Fixed effects, Bayes



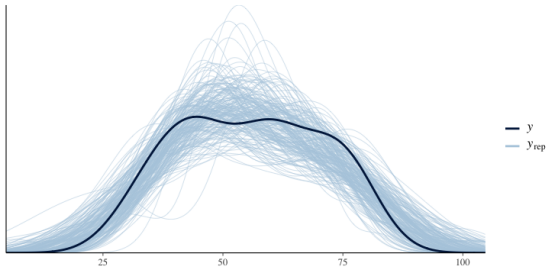
(d) Mixed effects, Bayes

Figure 11: Predictions by different models.

We can also perform visual posterior predictive check for Bayesian models with use of the bayesplot package. Below we compare density of response variable y (black line) with densities of generated y (200 posterior draws, blue lines). We can see that models simulations are relatively close to the y values.



(a) Fixed effects



(b) Mixed effects

Figure 12: Posterior predictive check.

6 Model comparison with cross-validation

In this section we will compare two Bayesian linear models: fixed effects vs mixed effects. We use leave-one-out cross-validation (LOO-CV) technique. The Bayesian LOO estimate of out-of-sample predictive fit is

$$\begin{aligned} elpd_{loo} &= \text{expected log pointwise predictive density for a new dataset} \\ &= \sum_{i=1}^{n} \log p(y_i|y_{-i}) \end{aligned} \tag{7}$$

where

$$\log p(y_i|y_{-i}) = \int p(y_i|\theta)p(\theta|y_{-i})d\theta \tag{8}$$

We will use **loo** R package, which computes PSIS-LOO CV, efficient approximation of leave-one-out (LOO) cross-validation for Bayesian models using Pareto smoothed importance sampling (PSIS). This is an implementation of the methods described in Vehtari, Gelman, and Gabry (2017) and Vehtari, Simpson, Gelman, Yao, and Gabry (2019).

The printed output from the loo function shows:

- the estimates of expected log predictive density $\hat{elpd}_{loo}$
- effective number of parameters $\hat{p}_{loo}$
- the LOO information criterion $-2\ \hat{elpd}_{loo}$

The estimates of expected log predictive density for both models are very similar (see Listing 6).

The reliability and approximate convergence rate of the PSIS-based estimates can be assessed using the estimates for the shape parameter k of the generalized Pareto distribution. Briefly, if all the $\hat{k}$-values are about 0.7, the PSIS-LOO estimate can be considered to be reliable, otherwise there is a concern that it may be biased (too optimistic, overestimating the predictive accuracy of the model).

Summary for each model's $\hat{k}$-values are under Pareto k diagnostic values section shows that for fixed effects model all Pareto k estimates are good ($\hat{k} < 0.5$), but for mixed effects model there are $\hat{k}$-values larger than 0.7.

Related code can be found in R directory of github repo.

7 Conclusions

In this report we used MLE and Bayesian approach to fit LMM. If we compare MLE and MAP estimates of fixed and mixed effects models, we can see that estimates are very close. Coefficients β_0 and β_1 are different from zero, while the rest of coefficients are close to zero. Thus, we can make a conclusion that only fixed factor Treatment reliably affects the relative amount of wakefulness as measured by ECoG and there are no Genotype or Treatment:Genotype effects.

Parameters	Fixed+MLE	Fixed+Bayes	Mixed+MLE	Mixed+Bayes
β_0 Intercept	44.56 ± 2.70	44.55 ± 2.75	44.11 ± 2.78	42.67 ± 4.34
β_1 Treatment	3.0 ± 0.48	2.96 ± 0.48	3.08 ± 0.46	3.42 ± 0.50
β_2 Genotype	-0.51 ± 3.90	-0.44 ± 3.93	-0.38 ± 3.95	0.25 ± 5.97
β_3 Treat:Gen	-0.09 ± 0.68	-0.10 ± 0.68	-0.17 ± 0.65	-0.43 ± 0.70

Table 1: Estimated model parameters.

Although PSIS-LOO estimation of Bayesian hierarchical model with varying intercept might be unreliable and we cannot conclusively check how this model performs compared to Bayesian fixed effect model, based on MSE errors for predicted values we might prefer to use it. Addition of variance related to repeated measures seems to result in a better fit to the data.

In this simple example, we showed that although all models had similar parameter estimates, it seems that mixed effects models are better at prediction as additional variance due to grouping factor is included in the model.

We could try out different priors and hyperpriors to get better model. Also, as there was indication that relationship between wake amount and predictors may not be linear, we could try to fit non-linear dose-response curve instead.

Finally, this example demonstrates the flexibility of Bayes LMM and suitability for analyses of hierarchical data which is very common in biomedical research. The disadvantage of this approach is that it involves somewhat more complex computations and theory. Besides, troubleshooting might be also more complex and requires thorough training in statistics.

8 Listings

Listing 1: Fixed effect model with lm()

```
lm(formula = WAKE ~ Treatment + Genotype + Treatment:Genotype,
    data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-15.261  -5.324   2.135   5.985  18.593

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          44.5675     2.7044  16.480 2.90e-16 ***
Treatment             2.9591     0.4776   6.195 9.33e-07 ***
Genotype1            -0.5100     3.8697  -0.132    0.896
Treatment:Genotype1  -0.0925     0.6759  -0.137    0.892
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.224 on 29 degrees of freedom
Multiple R-squared:  0.7195,    Adjusted R-squared:  0.6905
F-statistic: 24.79 on 3 and 29 DF,  p-value: 3.745e-08
```

Listing 2: Fixed effect model with RStan flat priors

```
data {
  int<lower=0> N;                              // n.o. samples
  real y[N];                                   // measurements, response var
  real treat[N];                               // predictor treatment
  int<lower=0,upper=1> gen[N];                 // predictor genotype
}

parameters {
  real<lower=0> sigma;     // sheared standard deviation, (0,+inf)
  vector[4] beta;          // intercept and slopes, (-inf,+inf)
}

model {
  real mu;
  for (i in 1:N){
      mu = beta[1] + beta[2]*treat[i] + beta[3]*gen[i] + beta[4]*treat[i]*gen[i];
      y[i] ~ normal(mu, sigma);
      }
}
```

	mean	se_mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
beta[1]	44.63	0.08	2.92	38.80	42.80	44.68	46.56	50.20	1326	1
beta[2]	2.94	0.01	0.51	1.96	2.60	2.94	3.28	3.95	1211	1
beta[3]	−0.57	0.11	4.17	−8.92	−3.26	−0.56	2.10	7.84	1491	1
beta[4]	−0.09	0.02	0.72	−1.53	−0.56	−0.07	0.40	1.30	1279	1

Listing 3: Fixed effect model with RStan weak priors

```
data {
  int<lower=0> N;                              // n.o. samples
  real y[N];                                   // measurements, response var
  real treat[N];                               // predictor treatment
  int<lower=0,upper=1> gen[N];                 // predictor genotype
}

parameters {
  real<lower=0> sigma;     // sheared standard deviation
  vector[4] beta;          // intercept and slopes
}

model {
  real mu;
  sigma ~ normal(0,10);      // weakly informative prior
  beta ~ normal(0,100);     // weakly informative prior

  for (i in 1:N){
      mu = beta[1] + beta[2]*treat[i] + beta[3]*gen[i] + beta[4]*treat[i]*gen[i];
      y[i] ~ normal(mu, sigma);
      }
}

generated quantities {
  real mu;
  real ypred[N];

  for(i in 1:N){
      mu = beta[1] + beta[2]*treat[i] + beta[3]*gen[i] + beta[4]*treat[i]*gen[i];
      ypred[i] = normal_rng(mu, sigma);
  }
}

4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

          mean se_mean sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
sigma     8.51    0.02 1.16   6.57   7.69   8.38   9.21  11.09  2201    1
beta[1]  44.44    0.07 2.85  38.83  42.62  44.41  46.24  50.36  1722    1
beta[2]   2.97    0.01 0.51   1.93   2.64   2.97   3.30   3.97  1778    1
beta[3]  -0.42    0.10 4.05  -8.50  -3.04  -0.27   2.22   7.53  1619    1
beta[4]  -0.10    0.02 0.73  -1.49  -0.59  -0.10   0.37   1.38  1723    1
```

Listing 4: Mixed effect model with lm()

```
Linear mixed model fit by REML ['lmerMod']

Formula: WAKE ~ Treatment + Genotype + Treatment:Genotype + (1 | id)
   Data: df

REML criterion at convergence: 221.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.8690 -0.5677  0.1858  0.6421  2.1511

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept)  9.926   3.151
 Residual             58.145   7.625
Number of obs: 33, groups:  id, 18

Fixed effects:
                   Estimate Std. Error t value
(Intercept)         44.1194     2.7813  15.863
Treatment            3.0834     0.4606   6.695
Genotype1           -0.3840     3.9456  -0.097
Treatment:Genotype1 -0.1707     0.6499  -0.263

Correlation of Fixed Effects:
            (Intr) Trtmnt Gntyp1
Treatment   -0.642
Genotype1   -0.705  0.452
Trtmnt:Gnt1  0.455 -0.709 -0.633
```

Listing 5: Mixed effect model with RStan weak priors

```
data {
  int<lower=0> N;                              // n.o. samples
  int<lower=0> J;                              // n.o. subj
  real y[N];                                   // measurements, response var
  int id[N];                                   // subject id
  real treat[N];                               // predictor treatment
  int<lower=0,upper=1> gen[N];                 // predictor genotype
}

parameters {
  real<lower=0> sigma;     // sheared standard deviation
  vector[4] beta;          // intercept and slopes
  vector[J] u;             // subj intercepts
}

model {
  real mu;
  sigma ~ normal(0,10);      // weakly informative prior
  beta ~ normal(0, 100);     // weakly informative prior
  u ~ normal(0,10);          // subj intercepts

  for (i in 1:N){
      mu = beta[1] + u[id[i]] +
      beta[2]*treat[i] + beta[3]*gen[i] + beta[4]*treat[i]*gen[i];

      y[i] ~ normal(mu, sigma);
      }
}

generated quantities {
  real mu;
  real ypred[N];

  for(i in 1:N){
       mu = beta[1] + u[id[i]] +
       beta[2]*treat[i] + beta[3]*gen[i] + beta[4]*treat[i]*gen[i];
       ypred[i] = normal_rng(mu, sigma);
  }
}

4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

          mean  se_mean  sd     2.5%     25%     50%     75%   97.5%  n_eff  Rhat
sigma     7.27    0.03   1.28    5.26    6.34    7.11    8.02   10.19   1717     1
beta[1]  42.69    0.12   4.34   34.43   39.69   42.65   45.56   51.34   1290     1
beta[2]   3.42    0.01   0.49    2.43    3.10    3.42    3.74    4.43   2208     1
beta[3]   0.28    0.17   5.96  -11.46   -3.88    0.20    4.43   11.79   1278     1
beta[4]  -0.44    0.01   0.68   -1.82   -0.88   -0.42    0.01    0.89   2130     1
```

Listing 6: PSIS-LOO

```
Output for Fixed Effects model:

Computed from 4000 by 33 log-likelihood matrix

          Estimate  SE
elpd_loo   -119.3 3.8
p_loo         4.2 1.0
looic       238.6 7.5
------
Monte Carlo SE of elpd_loo is 0.1.

All Pareto k estimates are good (k < 0.5).
See help('pareto-k-diagnostic') for details.


Output for Mixed Effects model:

Computed from 4000 by 33 log-likelihood matrix

          Estimate  SE
elpd_loo   -122.7 4.3
p_loo        15.6 2.8
looic       245.4 8.7
------
Monte Carlo SE of elpd_loo is NA.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)       8    24.2%   970
 (0.5, 0.7]   (ok)        20    60.6%   206
   (0.7, 1]   (bad)        4    12.1%   93
   (1, Inf)   (very bad)   1     3.0%   21
See help('pareto-k-diagnostic') for details.
```

9 References

- [1] "Statistics in a Nutshell: A Desktop Quick Reference", Sarah Boslaugh
- [2] "On the widespread and critical impact of systematic bias and batch effects in single-cell RNA-Seq data", Stephanie Hicks et al.
- [3] "Linear Mixed Effects Models" TensorFlow tutorial
- [4] "Linear Mixed Model from Scratch" blogpost by Nikolay Oskolkov
- [5] "Fitting Linear Mixed-Effects Models using lme4" D. Bates et al
- [6] stats.stackexchange forum
- [7] Gelman, A. (2006). Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper). Bayesian analysis, 1(3), 515-534.