# HW 1

## Question 1

Load the `Surrogate` package in R and load the dataset `Schizo_PANSS`:

```
library(Surrogate)
data("Schizo_PANSS")
```

The dataset combines five clinical trials aimed at determining if risperidone decreases the Positive and Negative Syndrome Score (PANSS) over time compared to a control treatment for patients with schizophrenia. These are longitudinal trials where patients are assessed at weeks 1, 2, 4, 6 and 8 after being assigned to a treatment arm.

Each row in the dataset is a different trial participant, and `Week1`, `Week2`, `Week4`, `Week6`, `Week8` records the change in PANSS from baseline. The variable `Treat` represents whether the patient was enrolled in the control (-1) or if the patient was in the risperidone arm (1).

Subset the data to include only `Week1`, `Week4` and `Week8`:

```
hw_data <- Schizo_PANSS[,c("Id","Treat","Week1","Week4","Week8")]
```

1. Summarize the missingness patterns in the `hw_data` dataset. How many missingness patterns are there? What are they? What proportion of patients are associated with each missingness pattern?
2. How would you assess whether there is evidence that treatment affects missingness? Is there evidence that treatment affects the missingness pattern?
3. How many people dropped out of the study after Week 1, or Week 4 vs. had intermittent missingness? For patients who dropped out, is there evidence that PANSS at the prior measurement predicted dropout? What about for the patients with intermittent missingness?
4. Is it reasonable to assume that missingness is MCAR for this dataset? Why or why not? What about MAR?
5. Subset the data to complete cases only and, using the algorithm we learned in class:

$$\beta^{(t+1)} = \left(\sum_i X_i^T (\Sigma^{(t)})^{-1} X_i\right)^{-1} \sum_i X_i^T (\Sigma^{(t)})^{-1} y_i$$

$$\Sigma^{(t+1)} = \frac{1}{n} \sum_i (y_i - X_i \beta^{(t)})(y_i - X_i \beta^{(t)})^T$$

Fit the following model to the complete case data:

$$y_i \mid \text{Treat}_i \sim \text{Normal}(\mu + \beta_1 \text{Treat}_i + \beta_2 t + \beta_3 \text{Treat}_i t, \Sigma)$$

where $t$ is the vector $1, 4, 8$ indicating at what time points the measurements were taken and $\mu$ is a scalar mean.

Include your MLEs for $\Sigma$ and the vector $(\mu, \beta_1, \beta_2, \beta_3)$, and be sure to interpret your inferred coefficients in the context of the PANSS dataset.

It'll help to reshape the data into long format from wide format:

```
comp_case <- hw_data |>
  subset(
    !is.na(Week1) &
      !is.na(Week4) &
      !is.na(Week8)
  )
long_case <-
  stats::reshape(
    comp_case,
    direction = "long",
    varying = 3:5,
    sep = ""
  )[,-5]
names(long_case) <- c("Id","Treat","time","panss")
```

In order to make sure your algorithm is successful, include a test of your algorithm on on this simulated dataset, where you compare your algorithm's inferences to the true values of $\beta$ and $\Sigma$:

```
set.seed(123)
n <- 10000
p <- 5
K <- 3
X <- list()
beta <- rnorm(p)
L <- matrix(rnorm(K * K),K,K)
Sigma <- L %*% t(L)
```

```r
y <- list()
for (i in 1:n) {
  X[[i]] <- matrix(rnorm(p * K),K,p)
  y[[i]] <- X[[i]] %*% beta + MASS::mvrnorm(mu = rep(0,K), Sigma = Sigma)
}
```

6. How might you expand this model based on your initial data analysis? There are no wrong answers.

## Question 2

In Question 1, you analyzed the complete cases in the PANSS meta-analysis dataset using the repeated measures model:

$$y_i \mid \text{Treat}_i \sim \text{Normal}(\mu + \beta_1 \text{Treat}_i + \beta_2 t + \beta_3 \text{Treat}_i t, \Sigma)$$

Let $\beta = (\mu, \beta_1, \beta_2, \beta_3)$ in what follows. Fit the following Bayesian model for the complete cases:

$$\begin{aligned} y_i \mid \text{Treat}_i, \beta, \Sigma &\sim \text{Normal}(\mu + \beta_1 \text{Treat}_i + \beta_2 t + \beta_3 \text{Treat}_i t, \Sigma) \\ \beta \mid \mu_0, \Sigma_0 &\sim \text{Normal}(\mu_0, \Sigma_0) \\ \Sigma \mid \nu_0, V_0 &\sim \text{Inv-Wishart}(\nu_0, V_0) \end{aligned}$$

The sampling algorithm is as follows:

1. Repeat the following for $b = 1, \ldots, B$ chains.
2. Draw $\beta^0, \Sigma^0$ starting values from the priors for $\beta$ and $\Sigma$.
3. For $t = 0, \ldots, S-1$

    a. Draw $\beta$ according to the distribution

$$\begin{aligned} \beta^{t+1} \mid \Sigma^t, y, X &\sim \text{Normal}(\bar{\beta}, (\textstyle\sum_i X_i^T (\Sigma^{(t)})^{-1} X_i + (\Sigma_0)^{-1})^{-1}) \\ \bar{\beta} &= (\textstyle\sum_i X_i^T (\Sigma^{(t)})^{-1} X_i + \Sigma_0^{-1})^{-1} (\textstyle\sum_i X_i^T (\Sigma^{(t)})^{-1} y_i + \Sigma_0^{-1} \mu_0) \end{aligned}$$

    b. Draw $\Sigma^{t+1}$ according to

$$\Sigma^{t+1} \mid \beta^{t+1}, y, X \sim \text{Inverse-Wishart}(n + \nu_0, V_0 + \textstyle\sum_i (y_i - X_i \beta^t)(y_i - X_i \beta^t)^T)$$

4. Discard the first $\frac{S}{2}$ iterations, and keep the set of draws $\{(\beta^s, \Sigma^s),\ s = \frac{S}{2}, \ldots, S-1\}$.

The final set of draws $B \times \frac{S}{2}$ draws are approximately distributed according to $p(\beta, \Sigma \mid y)$

## Part 1

Before running your model, test your algorithm on the simulated dataset above, using the prior hyperparameters, with $B = 4$ and $S = 500$:

$$\mu_0 = 0, \Sigma_0 = I_p, \nu_0 = 3, V_0 = I_p$$

In order to run this model, you'll need to be able to draw from an inverse Wishart distribution. Use the version that is implemented in the packages `MCMCpack`.

When you run your chains, you'll have 4 chains with 250 draws of each parameter per chain. You can use these draws to assess whether there is evidence against having reached the steady state distribution. Use the package `posterior` and `rhat` in the posterior package to report the univariate $\hat{R}$ statistics for each dimension of the beta vector.

To do so, you'll need to arrange the draws for each dimension of beta into a matrix with rows corresponding to the 250 iterations and columns corresponding to the chains. Then you can pass that matrix to `posterior::rhat` to get an estimate of how much the scale of the posterior distribution would shrink as $S \to \infty$. Numbers near 1, or $\hat{R} \leq 1.01$ indicate that one may not have much to gain from running the chains for more iterations.

Can you describe a better way to test whether your algorithm is correctly written using simulated data and posterior quantiles? See Talts et al. (2018) for some ideas.

## Part 2

Run your model with the same priors as above on the `comp_case` data from above, and report 95% central posterior quantiles for each dimension of $\beta$.

Also report the 95% posterior quantiles for the parameters: $\Sigma_{2,1}/\sqrt{\Sigma_{1,1}\Sigma_{2,2}}$, $\Sigma_{3,1}/\sqrt{\Sigma_{1,1}\Sigma_{3,3}}$, and $\Sigma_{3,2}/\sqrt{\Sigma_{2,2}\Sigma_{3,3}}$, or the correlation between the errors for each of the PANSS measurement occasions.

This may be done by defining the appropriate transformations of the parameter draws.

Talts, Sean, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. 2018. "Validating Bayesian Inference Algorithms with Simulation-Based Calibration." *arXiv Preprint arXiv:1804.06788*.