

STAT 547: Bayesian Workflow

Homework

Exercise 1 (pharmacokinetic model)

In this exercise, we will walk through the steps of Bayesian workflow to analyze clinical observations from a single patient. Our goal is to understand how quickly a drug compound is absorbed and cleared from the patient's body. The patient orally receives a drug dose of 1200 mg at time $t = 0$. We then measure the drug concentration (mg/L) in the patient's blood over time (hours), and obtain the following:

$$\begin{aligned} y &= (3.79, 5.80, 12.79, 15.52, 9.98, 18.65, 13.21, 13.91, 8.16, 4.81, 4.59, 2.23) \\ t &= (0.083, 0.167, 0.25, 0.5, 0.75, 1, 1.5, 2, 3, 4, 6, 8). \end{aligned}$$

A standard pharmacokinetic model is the *one-compartment* model which describes how the drug diffuses from the gut into the body before being cleared. This process is described by an ordinary differential equation,

$$\begin{aligned} u'_{\text{gut}}(t) &= -k_a u_{\text{gut}}(t), \\ u'_{\text{cent}} &= k_a u_{\text{gut}}(t) - \frac{CL}{V_{\text{cent}}} u_{\text{cent}}(t), \end{aligned} \tag{1}$$

with

- $u_{\text{gut}}(t)$: drug mass in the gut (mg),
- $u_{\text{cent}}(t)$: drug mass in the *central compartment* (mg), that is the blood and organs into which the drug diffuses rapidly,
- k_a : absorption rate constant (h^{-1}),
- CL : elimination clearance from the central compartment (L / h),
- V_{cent} : volume of the central compartment (L).

The patient receives the drug orally at time $t = 0$. This corresponds to a bolus dose in the gut and so the initial conditions of the differential equations are $u_{\text{gut}}(0) = 1200$ and $u_{\text{cent}}(0) = 0$.

The drug concentration is given by

$$c(t) = u_{\text{cent}}(t)/V_{\text{cent}}, \tag{2}$$

and the measurement model is

$$y(t) \sim \text{logNormal}(\log c(t), \sigma), \tag{3}$$

with σ an unknown standard deviation parameter. Hence, the parameters of our model are $\theta = (k_a, CL, V_{\text{cent}}, \sigma)$. Based on results on other patients, the following priors are available to us:

$$\begin{aligned} CL &\sim \text{logNormal}(\log 10, 0.25), \\ V_{\text{cent}} &\sim \text{logNormal}(\log 35, 0.25), \\ k_a &\sim \text{logNormal}(\log 2.5, 1) \\ \sigma &\sim \text{normal}^+(0, 1). \end{aligned} \tag{4}$$

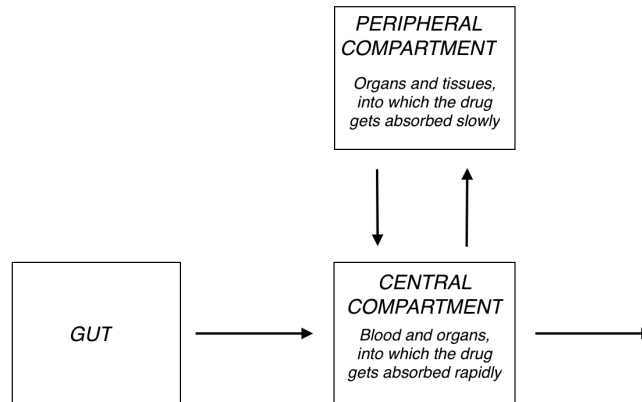


Figure 1: Pharmacokinetic model

- Implement this model in Stan. (*Tip*: You can use a numerical integrator to solve the ODE in eq.1. For faster inference, you can also resort to a matrix exponential or solve the ODE by hand.)
- Fit the model, check the quality of the inference (convergence, effective sample size), and perform posterior predictive checks.
- A more sophisticated pharmacokinetic model divides the body into a further compartment—the *peripheral compartment* into which the drug diffuses slowly. The differential equation for this model is

$$\begin{aligned}
 u'_{\text{gut}}(t) &= -k_a u_{\text{gut}}(t), \\
 u'_{\text{cent}}(t) &= k_a u_{\text{gut}}(t) - \left(\frac{CL}{V_{\text{cent}}} + \frac{Q}{V_{\text{cent}}} \right) u_{\text{cent}}(t) + \frac{Q}{V_{\text{peri}}} u_{\text{peri}}(t), \\
 u'_{\text{peri}}(t) &= \frac{Q}{V_{\text{cent}}} u_{\text{cent}}(t) - \frac{Q}{V_{\text{peri}}} u_{\text{peri}}(t),
 \end{aligned} \tag{5}$$

which introduces two new parameters:

- Q : intercompartmental clearance (L/h)
- V_{peri} : volume of the peripheral compartment (L).

For these parameters, we may use the following priors:

$$Q \sim \text{logNormal}(\log 15, 0.5); V_{\text{peri}} \sim \text{logNormal}(\log 105, 0.5). \tag{6}$$

Fit this model and perform posterior predictive checks. Then, compare the two pharmacokinetic models using (approximate) Bayesian leave-one-out cross-validation.

Exercise 2 (Hamiltonian Monte Carlo)

In this exercise, you will code an HMC algorithm and evaluate its performance with varying tuning parameters.

- (a) Write a function that runs “static” HMC. At each iteration, your code should simulate a Hamiltonian trajectory using a leapfrog integrator with L steps of size ϵ . The proposal is then accepted or rejected via a Metropolis procedure. Set the mass matrix to I (the identity matrix). Your function should accept the following arguments:
- `log_p`: a function that returns the target’s (unnormalized) log density.
 - `grad_log_p`: a function that returns the gradient of the target’s log density.
 - `theta0`: a starting point for your Markov chain.
 - `n_iter`: the total number of MCMC iterations.
 - `eps`: the step size of leapfrog integrator.
 - `L`: the number of leapfrog steps per iteration.

The output of your function should be MCMC samples.

Test your function on a one-dimensional standard normal and check that the mean of the MCMC samples approaches 0 and its variance approaches 1. (This should hold for any reasonable choice of `n_iter`, `eps`, and `L`.)

- (b) Consider now as a target distribution a 5-dimensional multivariate normal. You may generate a random covariance matrix or use the following one:

$$\Sigma = \begin{pmatrix} 2.82 & 2.71 & 0.180 & 2.42 & -1.87 \\ 2.71 & 7.35 & -4.02 & -1.17 & -1.05 \\ 0.180 & -4.02 & 8.85 & 1.64 & 3.81 \\ 2.42 & -1.17 & 1.64 & 5.94 & -3.95 \\ -1.87 & -1.05 & 3.81 & -3.95 & 6.40 \end{pmatrix}$$

Run your HMC algorithms on this target with $\epsilon = 0.05$, $L = 10$ and `n_iter` = 1000. Discard half of your samples as a warmup. Report the mean, median, standard deviation, 5th and 95th posterior quantiles. (You may use an existing package such as `posterior` in R or `Arviz` in Python to do this.)

- (c) Report \hat{R} and ESS. Show the trace plots and density plots. (Tip: to compute the convergence diagnostics, you should run at least 4 independent Markov chains.)
- (d) Create a plot showing the squared error of your Monte Carlo estimate of the mean for the first dimension as a function of the total length of the Markov chains. As before, discard the first half of the samples as a warmup. Create the same plot, this time for a Monte Carlo estimate of the second moment.
- (e) We will now experiment with different trajectory lengths. In their analysis of the Gaussian case, Hoffman et al. [2021] prescribe a trajectory length which verifies,

$$L^* \epsilon = \lfloor 2.25 \max_i \sqrt{\Sigma_{ii}} \rfloor. \quad (7)$$

Run your sampler with $L = 80$ and $L = L^*$. For each choice of L (including $L = 10$), report \hat{R} , ESS and ESS per gradient evaluation.

- (f) Rather than keeping L fixed, it is often beneficial to jitter L at each iteration. Specifically, at each iteration we draw L uniformly from $\{1, 2, \dots, L_{\max}\}$. Run HMC using $L_{\max} = L^*$ and report \hat{R} and the ESS per gradient evaluation.

Exercise 3 (variational inference)

In this exercise, we will do a detailed analysis of the case where we use VI to approximate a non-factorized Gaussian with a factorized Gaussian. As our variational objective, we chose the reverse Kulback-Leibler divergence,

$$\text{KL}(q||p) = \int (\log q(z) - \log p(z))q(z)dz. \quad (8)$$

- (a) Show that $\text{KL}(q||p) \geq 0$ and $\text{KL}(q||p) = 0$ if and only if $p = q$.
- (b) Consider the case where p is a d -dimensional Gaussian with mean μ and non-diagonal covariance matrix Σ , and q is a d -dimensional Gaussian with mean ν and diagonal covariance matrix Ψ . Show that,

$$\text{KL}(q||p) = K + \frac{1}{2}(\nu - \mu)^T \Sigma (\nu - \mu) - \frac{1}{2} \log |\Psi| + \frac{1}{2} \sum_{i=1}^d [\Sigma^{-1}]_{ii} \Psi_{ii}, \quad (9)$$

where K is a constant which does not depend on the variational parameters ν and Ψ , and $\log |\Psi|$ denotes the log determinant of Ψ .

- (c) Show that the variational parameters which minimize $\text{KL}(q||p)$ are $\nu = \mu$ and $\Psi_{ii} = 1/[\Sigma^{-1}]_{ii}$.
- (d) We will now show that the variational approximation underestimates the variance of p . There exists several ways to do this. I'll provide guidance on one approach, however you should feel free to use another approach.
- (i) Show that $\text{Var}(z_i | z_{-i}) = 1/[\Sigma^{-1}]_{ii}$.
- (ii) Hence, show that $\Psi_{ii} \leq \Sigma_{ii}$ and that this inequality must be strict for at least two coordinates i if Σ is non-diagonal.

References

M. Hoffman, A. Radul, and P. Sountsov. An adaptive-mcmc scheme for setting trajectory lengths in hamiltonian monte carlo. In *International Conference on Artificial Intelligence and Statistics*, volume 130, pages 3907–3915, 2021.