Statistical Methods 2

Portfolio 8 – Metropolis-Hastings algorithm, Gibbs sampler and Metropolis-within-Gibbs

Complete the following task and submit your work on Blackboard by 4pm on Friday 28/04/2023

Consider the Pima Indians Diabetes dataset[1] $\{(y_i^0, x_i^0)\}_{i=1}^n$, for which y_i^0 indicates whether or not patient i has diabetes and where x_i^0 is a vector containing $p = 9$ diagnostic measurements. (Remark: If you want you can use another dataset on which a logistic regression model can be fitted.)

Assuming that $y_i^0 = 1$ if patient i has diabetes, and $y_i^0 = 0$ otherwise, we consider the logistic regression model

$$\Pr_{\alpha,\beta}^{(i)}(Y_i^0 = 1) = \frac{1}{1 + e^{-\alpha - \beta^\top x_i^0}}, \quad i = 1, \dots, n.$$

Let

$$L_n(\alpha, \beta) = \prod_{i=1}^n \Pr_{\alpha,\beta}^{(i)}(Y_i^0 = y_i^0)$$

be the likelihood function of the mode, $\pi(\alpha, \beta)$ be a prior distribution on $\mathbb{R}^{p+1}$ and

$$\pi(\alpha, \beta | y^0) \propto L_n(\alpha, \beta)\pi(\alpha, \beta) \tag{1}$$

be the resulting posterior distribution of (α, β) given the data.

Use a Metropolis-Hastings algorithm to approximate $\pi(\alpha, \beta|y^0)$. More precisely, you need

1. To choose a proposal distribution Q.
2. To implement **yourself** the Metropolis-Hastings algorithm that uses the proposal distribution Q chosen in 1 and targets $\pi(\alpha, \beta|y^0)$.
3. To assess the convergence of the algorithm (using the acceptance rate, trace plots and autocorrelation functions).
4. If needed, modify Q until the convergence behaviour of the algorithm is satisfactory.
5. Plot the estimated marginal posterior distribution for each of the 9 parameters of the model.

Possible choice for Q: Recall that the mapping $(\alpha, \beta) \mapsto \log L_n(\alpha, \beta)$ is concave. Hence, if the prior distribution has a log-concave density, the mapping

$$(\alpha, \beta) \mapsto \log \pi(\alpha, \beta | y^0)$$

is concave as well. In this case, it often works well in practice to choose $Q(z, \mathrm{d}z') = \mathcal{N}_{p+1}(z, c\boldsymbol{\Sigma}_n)$, with $c > 0$ a tuning parameter and where

$$\mu_n = \mathrm{argmax}_{(\alpha,\beta)\in\mathbb{R}^{p+1}} \log \pi(\alpha, \beta|y^0), \quad \boldsymbol{\Sigma}_n = -\big(\boldsymbol{H}_n(\mu_q)\big)^{-1}$$

[1] Available from the `R` package `mlbench`.

with

$$\boldsymbol{H}_n(\theta) = \Big(\frac{\partial^2}{\partial\theta_j \partial\theta_l} \log \pi(\theta|y^0)\Big)_{j,l=1}^{p+1}, \quad \forall \theta \in \mathbb{R}^{p+1}.$$

Remark that the $\mathcal{N}_{p+1}(\mu_n, \boldsymbol{\Sigma}_n)$ distribution is a Gaussian approximation of $\pi(\alpha, \beta|y^0)$, and therefore the matrix $\boldsymbol{\Sigma}_n$ can be interpreted as an estimate of the covariance matrix of (α, β) under $\pi(\alpha, \beta|y^0)$.