Brief of Baye’s Theorem and JAGS Package Plan

From Baye’s theorem video, I know that the prior is a condition *x* we know, and the posterior is what we know about *x* with the observation *y*. For example, there is a prior belief --you had heard from the neighbour that this week may not rain, and the prior belief is that today will be 30% probability to be rainy. However, there is a weather APP telling you today may have 75% probability of rain, and this new data will update your belief. Then, combining with the prior belief and the new data, we can get a posterior distribution, which considers there is 60% rain today.

JAGS (Just Another Gibbs Sampler) is an independent programme to implement MCMC simulations (Markov Chain Monte Carlo). It allows the Bayesian models (prior and likelihood) and external data as the input. Loading data from the file requires setting parameters of statistical information (e.g. sample size, number of successes)

After we get the true abundance as a matrix, we will implement the model estimation with for loop in ‘model’ block to define the probability model for the true abundance based on the multivariate normal distribution, then we can get the parameters: mean, variance-covariate, theta, and so on. Then we can calculate a likelihood from the ‘dinterval’ distribution to reinforce a posteriori restriction. Next, we can create a data list of the data from our generative simulation model; and plug those parameters and data list into ‘jags.model’.

The term “burn-in” describes the first stage of the MCMC sampling procedure, which enables the chain to explore the parameter space and reach a converged state. The generated samples are not used for inference during the burn-in phase. Burn-in is used to give the chains enough time to move from the initial starting point to a region of the parameter space where there are more likely to represent samples from the true posterior distribution. (The output of JAGS are samples from a posterior distribution)