Question 1 (a) Assuming that we know $\beta_0 = 5$, $\lambda = 0.4$, and $\sigma_i = 1, \quad \forall i$ and taking the prior on β_1 to be $N(0, sd = 10)$ the posterior distribution of β_1 up to a normalizing constant can be written as

$$\pi(\beta_1|\mathbf{Y}, X) \propto 0.2^n exp\left\{0.2\left(10.4n + 2\beta_1\sum_{i=1}^{n} x_i - 2\sum_{i=1}^{n} y_i\right) - \frac{\beta_1^2}{200}\right\}\left[\prod_{i=1}^{n} erfc(\frac{5.4 + \beta_1 x_i - y_i}{\sqrt{2}})\right]$$

I used a $N(\beta_1^{(t)}, \tau^2)$ as my proposal distribution. That is, the proposal density of β_1' conditional on the current state of the chain, $\beta_1^{(t)}$, is $q(\beta_1'|\beta_1^{(t)}) = \frac{1}{\sqrt{2\pi\tau^2}} exp\left\{\frac{-(\beta_1' - \beta_1^{(t)})}{2\tau^2}\right\}$. Where τ^2 is a tuning parameter, which was chosen by first searching over a broad and then increasingly narrower sets of values to optimize autocorrelation decay and maximize ESS. The optimal τ was chosen to be 0.75. Since this proposal is symmetric ($q(a|b) = q(b|a)$), we have Metropolis updates. Initial values for β_1 were first chosen arbitrarily over a broad range, and based on the convergence of these initial trials to 7.34, the initial value was then chosen to be 7.34 so as to decrease the time necessary for the chain to approach stationary behavior. To improve computational stability, all calculations were done on log scales.

The M-H algorithm

1. Initialize $\beta_1^{(0)}$ to 7.34
2. For $t = 0$ to $N - 1$ iterate
   (a) Generate $\beta_1' \sim q(\beta_1'|\beta_1^{(t)})$
   (b) Generate $U \sim Uniform(0, 1)$
   (c) Let

$$\beta_1^{(t+1)} = \begin{cases} \beta_1', & \text{if } log(U) < \sum_{i=1}^{n}\left[log(f(y_i; \beta_0, \beta_1', \sigma_i, \lambda)) - log(f(y_i; \beta_0, \beta_1^{(t)}, \sigma_i, \lambda))\right] + \frac{\beta_1^{(t)2}}{200} - \frac{\beta_1'^2}{200} \\ \beta_1^{(t)}, & \text{otherwise} \end{cases}$$

Question 1 (b) My estimate of $E_{\pi(\beta_1|\mathbf{Y},\mathbf{X})}(\beta_1)$ and associated MCMCse from a simulation of this algorithm are given in table 1 below.

Estimate	MCMCse
7.340965	0.001458387

Table 1: Table of MCMC Estimate of $E_{\pi(\beta_1|\mathbf{Y},\mathbf{X})}(\beta_1)$ and associated MCMCse

Question 1 (c) 95% Credible Interval Estimate for β_1 ($P(\beta_1 \in (L, B)|\mathbf{Y}, \mathbf{X}) = 0.95$) based on 2.5% and 97.5% sample quartiles of MCMC samples: $(\hat{L}, \hat{B}) = (6.716343, 7.933497)$

Question 1 (d) Smoothed Density Plot

From figure 1 we can see good convergence behavior of the M-H sampler, as the density estimate after N/2 samples and N samples align well.

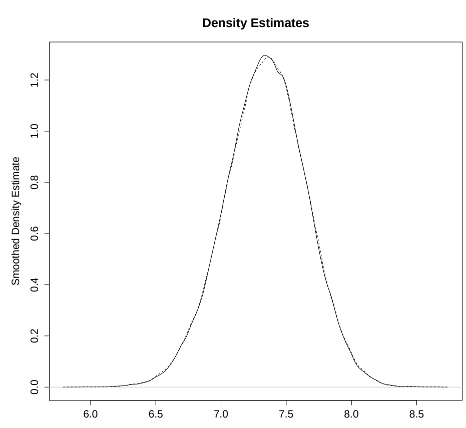

Figure 1: Smoothed density plot estimate of posterior density of β_1 after 100,000 (dashed line) and 200,000 (solid line) iterations.

Question 1 (e) To determine whether my approximations were accurate, I checked several diagnostics: trace plots for mixing behavior, sample autocorrelation, convergence of mean estimates, decay of MCMCse, comparison of density plots after 100,000 iterations and 200,000 iterations (see part d) and convergence of estimates from various starting points.

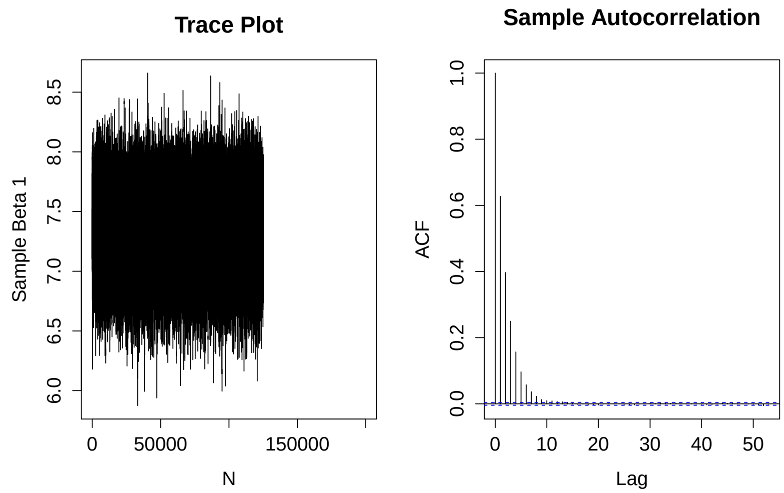

Figure 2: (left) Trace plot (right) Autocorrelation of samples

The trace plot in figure 2 shows good mixing behavior: we do not see the samples jumping and then sticking in different regions. The sharp decay in the autocorrelation plot in figure 2 also suggests that the algorithm is performing at reasonable efficiency as compared to the untuned sampler (plots of untuned sampler not shown here). Because of this sharp decay, in $N = 200,000$ runs (Acceptance rate: 0.438) the effective sample size was about 45,566 - so our estimator should have low variance.

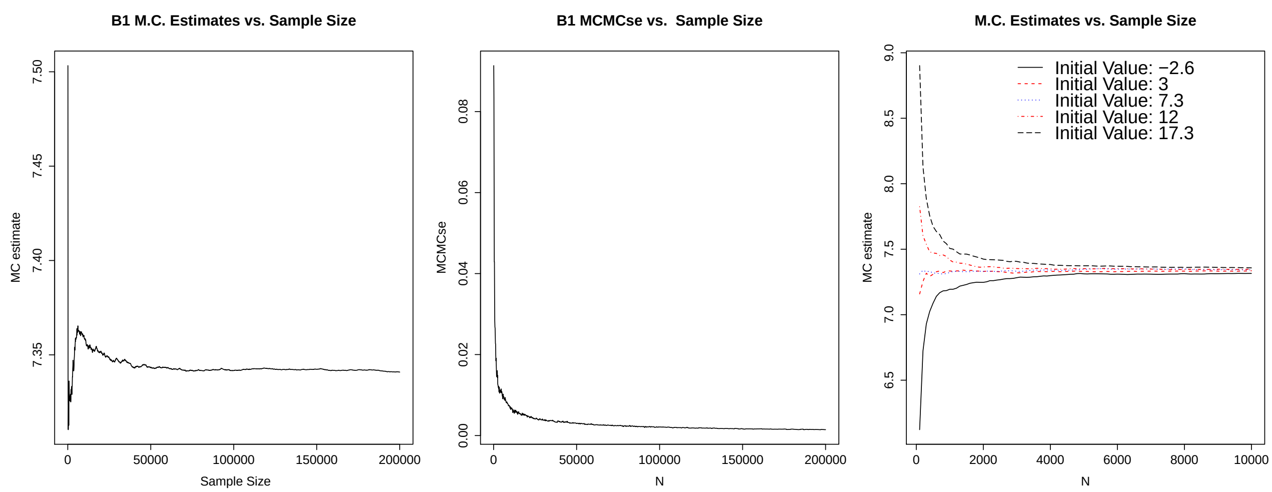

Figure 3: (left)MCMC estimates by sample size (center) MCMCse by sample size (right) convergence of estimates from various starting points (-2.6, 3, 7.3, 12, 17.3)

We can see from 3 that our estimates appear to converge from the plot of estimates against sample size. After about 50,000 samples the estimate of the posterior expectation of β_1 does not appear to jump around very much. We also see nice behavior in the decay of the MCMCse with increasing sample size. Finally, I checked several different starting locations (however, only 5 different locations are plotted), and we see convergence towards the same estimate, which provides some evidence that we have not missed other important modes of the posterior.

Question 2 (a) Initially I had used a variable at a time M-H algorithm to estimate the posterior expectations, and from those initial simulations I found strong correlation between the estimates of β_0 and β_1 (around -0.76), moderate correlation between β_0 and λ (around 0.4) and very low correlation between β_1 and λ. Because of the two stronger correlations, I used an all-at-once version of the M-H algorithm. Also, because λ has strictly positive support, it was more natural to consider the transformation $\phi = log(\lambda)$ in order to restrict the support of proposed λ to be positive. I used a multivariate normal $N_3(\boldsymbol{\theta}^{(t)}, \Sigma)$ proposal where $\boldsymbol{\theta}^{(t)} = (\beta_0^{(t)}, \beta_0^{(t)}, \phi^{(t)})'$,

$$\Sigma = \begin{pmatrix} \tau_0^2 & -0.76\tau_0\tau_1 & 0.42\tau_0\tau_2 \\ -0.76\tau_0\tau_1 & \tau_1^2 & 0.01\tau_1\tau_2 \\ 0.42\tau_0\tau_2 & 0.01\tau_1\tau_2 & \tau_2^2 \end{pmatrix}$$

and τ_0, τ_1, τ_2 are tuning parameters, which were chosen by considering different triplets on a $(0, 2) \times (0, 2) \times (0, 2)$ cube. The optimal τ settings were chosen to maximize ESS and optimize autocorrelation decay. These were selected to be

$\tau_0 = 0.2, \tau_1 = 0.3, \tau_2 = 0.1$. That is, I simulate $\boldsymbol{\theta}'|\boldsymbol{\theta}^{(t)} \sim N_3(\boldsymbol{\theta}^{(t)}, \Sigma)$. As in the first problem, at first many starting values were considered, and once some convergence behavior to the ones listed below was observed, the initial values were then set to $\beta_0^{(0)} = 2.346, \beta_1^{(0)} = 3.46, \lambda^{(0)} = 0.803$ in order to minimize the number of iterations for the chain to demonstrate stationary behavior. Here, the posterior can be written as

$$\pi(\beta_0, \beta_1, \phi|\mathbf{Y}, X) \propto \left(\frac{exp(\phi)}{2}\right)^n exp\left\{\frac{exp(\phi)}{2}\left(2(n\beta_0 + \beta_1\sum_{i=1}^n x_i) + exp(\phi)\sum_{i=1}^n \sigma_i^2 - 2\sum_{i=1}^n y_i\right)\right\} \cdot$$
$$\cdot \left[\prod_{i=1}^n erfc\left(\frac{\beta_0 + \beta_1 x_i + exp(\phi)\sigma_i^2 - y_i}{\sqrt{2}\sigma_i}\right)\right] exp\left\{\frac{-(\beta_0^2 + \beta_1^2)}{200} + \frac{\phi - e^\phi}{100}\right\} \quad (1)$$

Let the right-hand-side of the above be denoted by $h(\beta_0, \beta_1, \phi|\mathbf{Y}, X)$

Let $\boldsymbol{\theta} = (\beta_0, \beta_1, \phi)'$. Here again, since the proposal is symmetric, we have Metropolis updates.

All-at-Once M-H algorithm

1. Initialize $\beta_0^{(0)} = 2.346, \beta_1^{(0)} = 3.46, \lambda^{(0)} = 0.803$ (i.e. $\phi^{(0)} = log(\lambda^{(0)}) = log(0.803)$)
2. For $t = 0$ to $N - 1$ iterate
   (a) Generate $\boldsymbol{\theta}' \sim N_3(\boldsymbol{\theta}^{(t)}, \Sigma)$
   (b) Generate $U \sim Uniform(0, 1)$
   (c) Let

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\theta}', & \text{if } log(U) < log(h(\beta_0', \beta_1', \phi'|\mathbf{Y}, X)) - log(h(\beta_0^{(t)}, \beta_1^{(t)}, \phi^{(t)}|\mathbf{Y}, X)) \\ \boldsymbol{\theta}^{(t)}, & \text{otherwise} \end{cases}$$

Question 2 (b) The table below gives the posterior mean estimates, corresponding MCMCse, and 95% credible intervals for the posterior expectation of β_0, β_1, and λ

	β_0	β_1	λ
Estimate (MCMCse)	2.3467 (1.015x10^{-3})	3.4596 (1.499x10^{-3})	0.8037 (4.544x10^{-4})
95% Cred. Int	(2.0723, 2.6149)	(3.0379, 3.8774)	(0.6932, 0.9288)

Table 2: Table of MCMC Estimates, associated MCMCse, and 95% Credible Interval Estimates

Question 2 (c) The correlation between β_0 and β_1 based on the MCMC samples was estimated to be $\hat{\rho} = -0.7789$

Question 2 (d) Smoothed Density Plots Figure

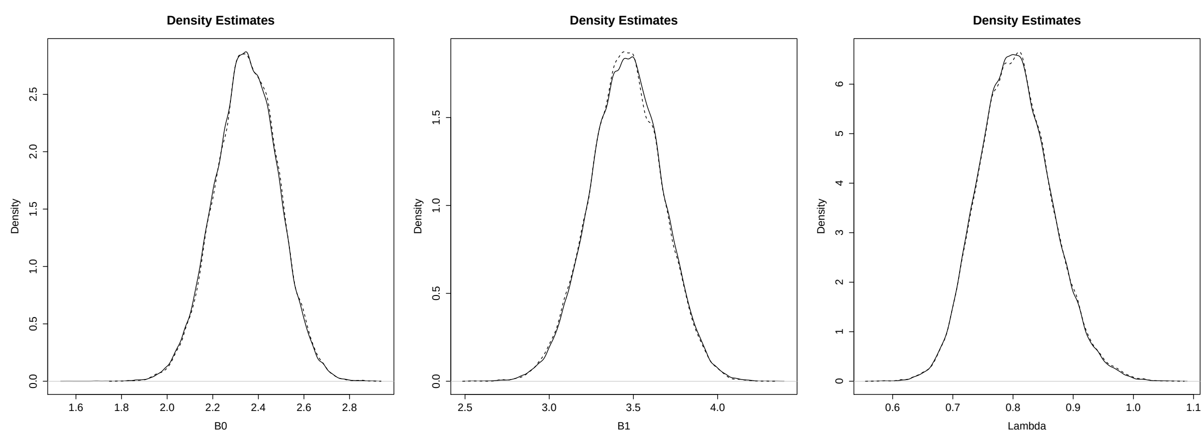

Figure 4: Approximate density plots for β_0 (left), β_1 (center), and λ (right) after 100,000 runs (dashed line) and 200,000 runs (solid line)

The estimated density plots for β_0, β_1 and λ after 100,000 and 200,000 runs all align well, suggesting good convergence behavior of the M-H algorithm.

Question 2 (e)

From the estimate against sample size plot in figure 3, we can see that the estimates for all three parameters appear to be converging. That is they are not jumping around as N increases. We also see good behavior in the estimate of the MCMCse of our estimates. That is, for all three parameters, the MCMCse decays with increasing sample size. The autocorrelation plots in figure 3 also appear to decay quickly, which indicates that our algorithm is not too sticky, and that the ergodic theorem should still apply. From $N = 200,000$ runs the effective sample size for estimating posterior expectations for each parameter was 18,668, 18,562,and 17,400 for samples of β_0, β_1, and λ respectively, which should be sufficiently large with which to do reliable estimation. The acceptance rate for the algorithm was 0.30949.

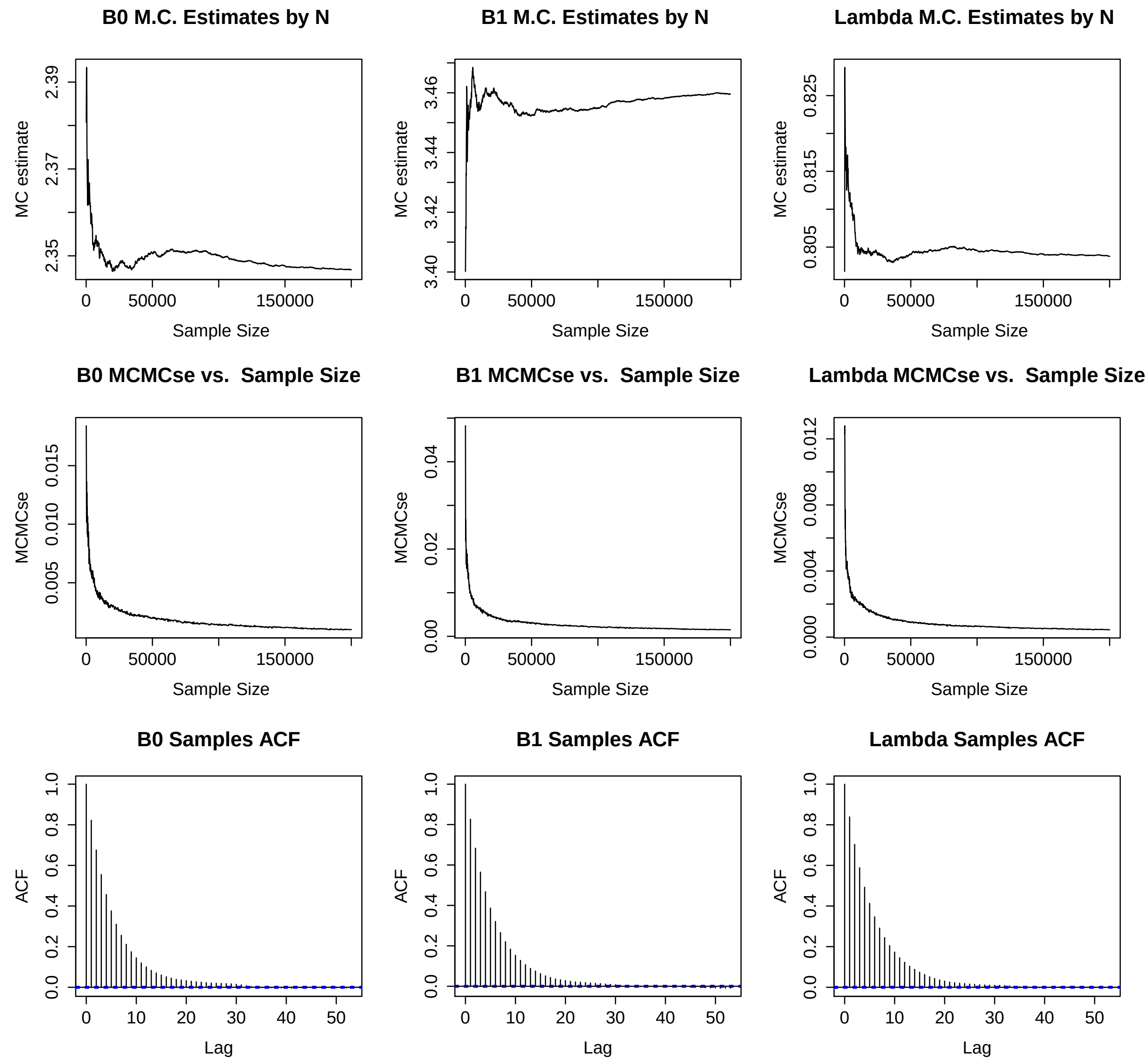

Figure 5: Diagnostic plots for reliability of MCMC estimates. Corresponding to β_0 (left), β_1 (center), and λ (right). MCMC estimates by sample size (top), MCMCse estimates by sample size (center), and sample autocorrelation (bottom)

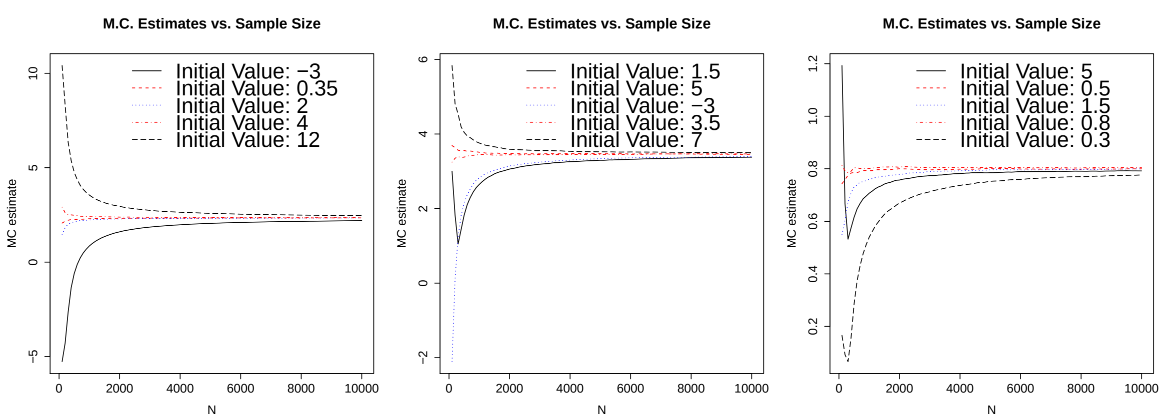

Figure 6: Convergence of posterior expectation estimates from 5 different starting values triplets ordered $(\beta_0, \beta_1, \lambda)$): (-3,1.5,5),(0.35,5,0.5),(2,-3,1.5),(4,3.5,0.8),(12,7,0.3)

Question 3 (a) Table 3 shows the estimates MCMCse and 95% credible intervals for the posterior expecation of the parameters. The same diagnostics as in question 2 were performed to assess the accuracy of the estimates and convergence of the MCMC algorithm and similar convergence was observed. Based on 200,000 runs the effective sample sizes were 21,776, 23,231, and 25,813 for posterior expectation estimates of β_0, β_1, and λ respectively.

	β_0	β_1	λ
Estimate (MCMCse)	0.1483 (1.159x10^{-3})	2.4741 (1.873x10^{-3})	0.1611 (3.513x10^{-5})
95% Cred. Int	(-0.1778,0.4635)	(1.9354, 3.0165)	(0.1506, 0.1723)

Table 3: Table of MCMC Estimates, associated MCMCse, and 95% Credible Interval Estimates

Question 3 (b) Figure 7 of smoothed density plots shows good convergence properties of the algorithm, as the density estimates after 100,000 runs and after 200,000 runs are nearly identical.

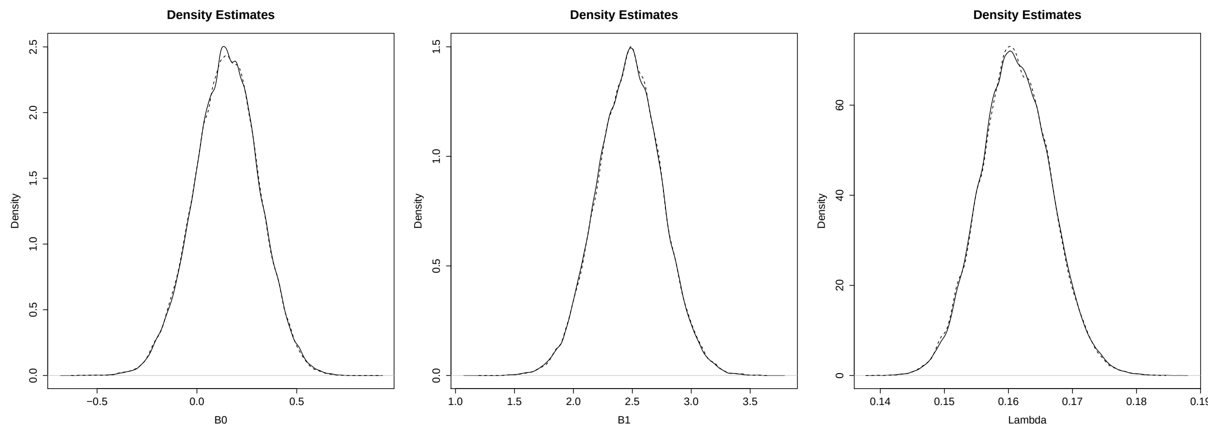

Figure 7: Approximate density plots for β_0 (left), β_1 (center), and λ (right) after 100,000 runs (dashed line) and 200,000 runs (solid line)

Question 3 (c) The first thing that I did before attempting the problem was to look at a scatter plot of x and y, and I noticed that there are two distinct clusters, with similar slopes but different intercepts which made me think that the model may be mispecified (i.e. perhaps the two groups either have different intercept terms or different rate parameters). Because of this, and being constrained to this model, I spent more time considering different starting values for the chain in case the posterior was multimodal, however regardless of my starting values, I observed convergence to the same final estimates. I also noticed that the correlation between my estimates of β_0 and λ samples was not as strong as in question 2. The estimated correlation between β_0 and β_1 samples was around -0.8. So instead of an all-at-once M-H approach, I blocked β_0 and β_1 together and blocked λ separately in my update steps. I took as my proposal for $(\beta_0, \beta_1)^T$ a $N_2((\beta_0^{(t)}\beta_1^{(t)})^T, \Sigma)$, where

$$\Sigma = \begin{pmatrix} \tau_0^2 & -0.8\tau_0\tau_1 \\ -0.8\tau_0\tau_1 & \tau_1^2 \end{pmatrix}$$

For ϕ (where $\phi = log(\lambda)$ - and I transform back after sampling), I used a $N(\phi^{(t)}, \tau_2^2)$ as my proposal. The optimal τ_0, τ_1, τ_2 were chosen again to maximize ESS and optimize the autocorrelation decay among the subsequent samples, and were ultimately chosen to be 0.325, 0.569 and 0.143 respectively. So the two full conditionals $\pi(\beta_0, \beta_1, |\mathbf{Y}, X, \phi)$ and $\pi(\phi|\mathbf{Y}, X, \beta_0, \beta_1)$ up to their normalizing constants, were the same as that which were used in the algorithm were the as the right hand side of equation 1. Call these two functions $h(\beta_0, \beta_1, |\mathbf{Y}, X, \phi)$ and $h(\phi|\mathbf{Y}, X, \beta_0, \beta_1)$. Here again, since the proposals are symmetric, we have Metropolis updates. The algorithm I used for this question is then

M-H algorithm

1. Initialize $\beta_0^{(0)} = 0.1483, \beta_1^{(0)} = 2.4741, \lambda^{(0)} = 0.1611$ (i.e. $\phi^{(0)} = log(\lambda^{(0)}) = log(0.1611)$)
2. For $t = 0$ to $N-1$ iterate
   (a) Generate $(\beta_0', \beta_1')^T \sim N_2(\beta_0^{(t)}, \beta_1^{(t)})^T, \Sigma)$
   (b) Generate $U \sim Uniform(0, 1)$
   (c) Let

$$(\beta_0^{(t+1)}, \beta_1^{(t+1)})^T = \begin{cases} (\beta_0', \beta_1')^T, & \text{if } log(U) < log(h(\beta_0', \beta_1'|\mathbf{Y}, X, \phi^{(t)})) - log(h(\beta_0^{(t)}, \beta_1^{(t)}|\mathbf{Y}, X, \phi^{(t)})) \\ (\beta_0^{(t)}, \beta_1^{(t)})^T, & \text{otherwise} \end{cases}$$

   (d) Generate $\phi' \sim N_(\phi^{(t)}, \tau_2^2)$
   (e) Generate $U \sim Uniform(0, 1)$
   (f) Let

$$\phi^{(t+1)} = \begin{cases} \phi', & \text{if } log(U) < log(h(\phi'|\mathbf{Y}, X, \beta_0^{(t+1)}, \beta_1^{(t+1)})) - log(h(\phi^{(t)}|\mathbf{Y}, X, \beta_0^{(t+1)}, \beta_1^{(t+1)})) \\ \phi^{(t)}, & \text{otherwise} \end{cases}$$